**CAPSTONE PROJECT-2**

**WEBSITE ORCHESTRATION**

**DONE BY**

**RANJITH KUMAR**

**Problem Statement:**

How to orchestrate a website with lesser time and higher availability along with Auto Scaling.

**Topics:**

In this AWS project, you have to deploy a high-availability PHP application withan external Amazon RDS database to Elastic Beanstalk. Running a DB instanceexternal to Elastic Beanstalk decouples the database from the lifecycle of your environment. This lets you connect to the same database from multiple environments, swap one database for another, or perform a blue/green

deployment without affecting your database.

**Highlights:**

Launch a DB instance in Amazon RDS

Create an Elastic Beanstalk Environment

Configure Security Groups and Scaling

**I have already created a VPC,Subnets,routetable,Internetgateway and associated it**.

STEP 1 : Open RDS from aws services

STEP 2 : Databases > Create Database

STEP 3 : Select standard,MYSQL,(As a free tier multi AZ not available),DB name and password and necessary information needed to create a DB instance

STEP 4 : Db instance has been created

STEP 5 : Now select the db instance and select the security group and allow inbound source as mysql port

STEP 6 : Open Elastic Beanstalk

STEP 7 : Environment > Create environment

STEP 8 : Give name, platform as PHP,service and instance profile role,VPC,subnets,Root volume,security groups and necessary information needed to create

STEP 9 : Environment has been created

STEP 10 : Select the environment > configuration choose the security group to attach to the binstances, in addition to the instance security group that Elasticbeanstalk creates.

STEP 11 : Select Configuration> Instance traffic & scaling. Edit

the Environment propertiessection, define the variables that your application reads to construct a connection string. For compatibility with environments that have an integrated RDS DB instance, use the

following:-

**RDS\_HOSTNAME**

**RDS\_PORT**

**RDS\_DB\_NAME**

**RDS\_USERNAME**

**RDS\_PASSWORD**

STEP 12 : Select Configuration > Instance traffic & scaling. In the autoscaling section minimum instance to 2

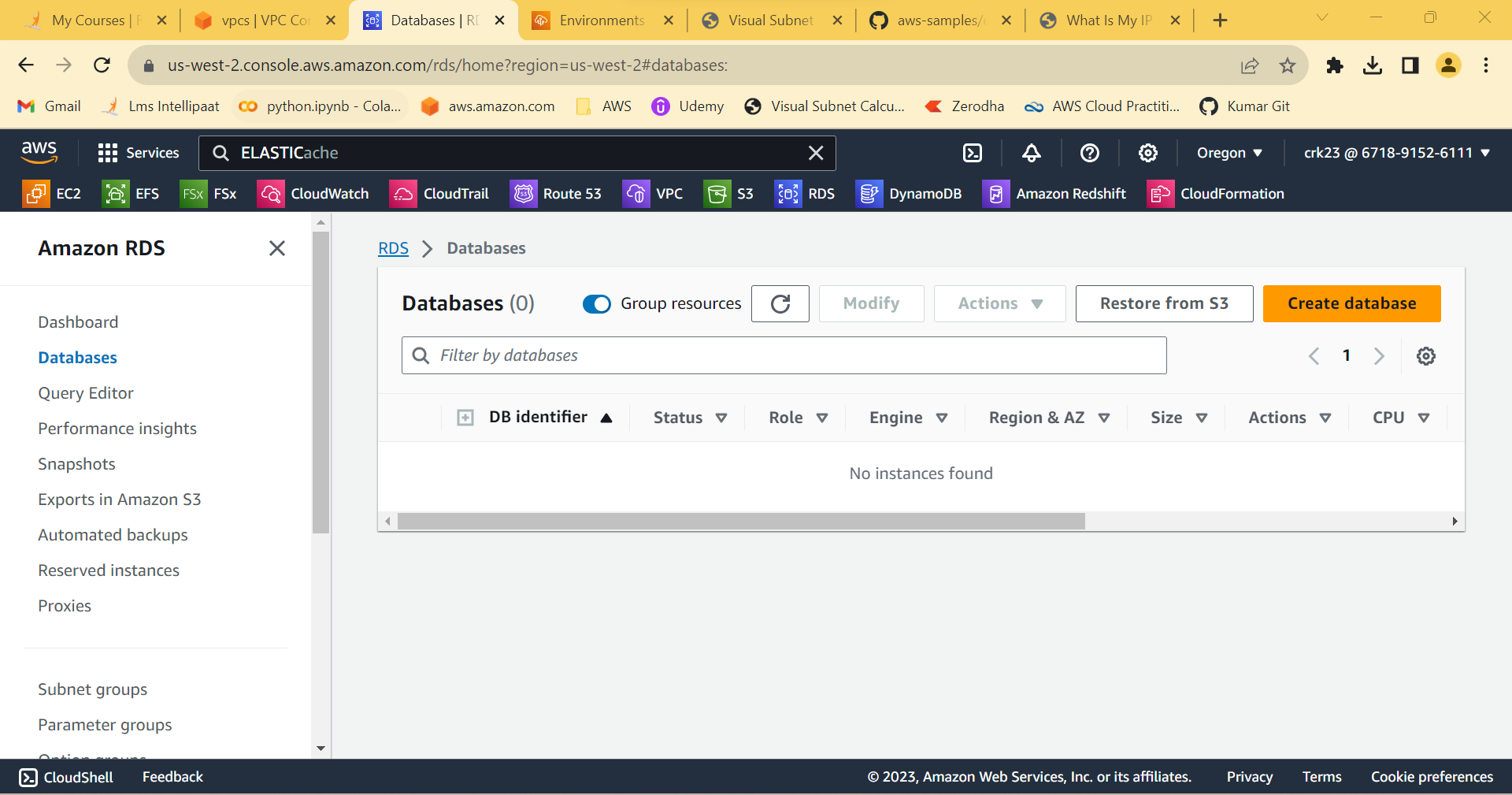
STEP 13 : Open elastic beanstalk and select the environment

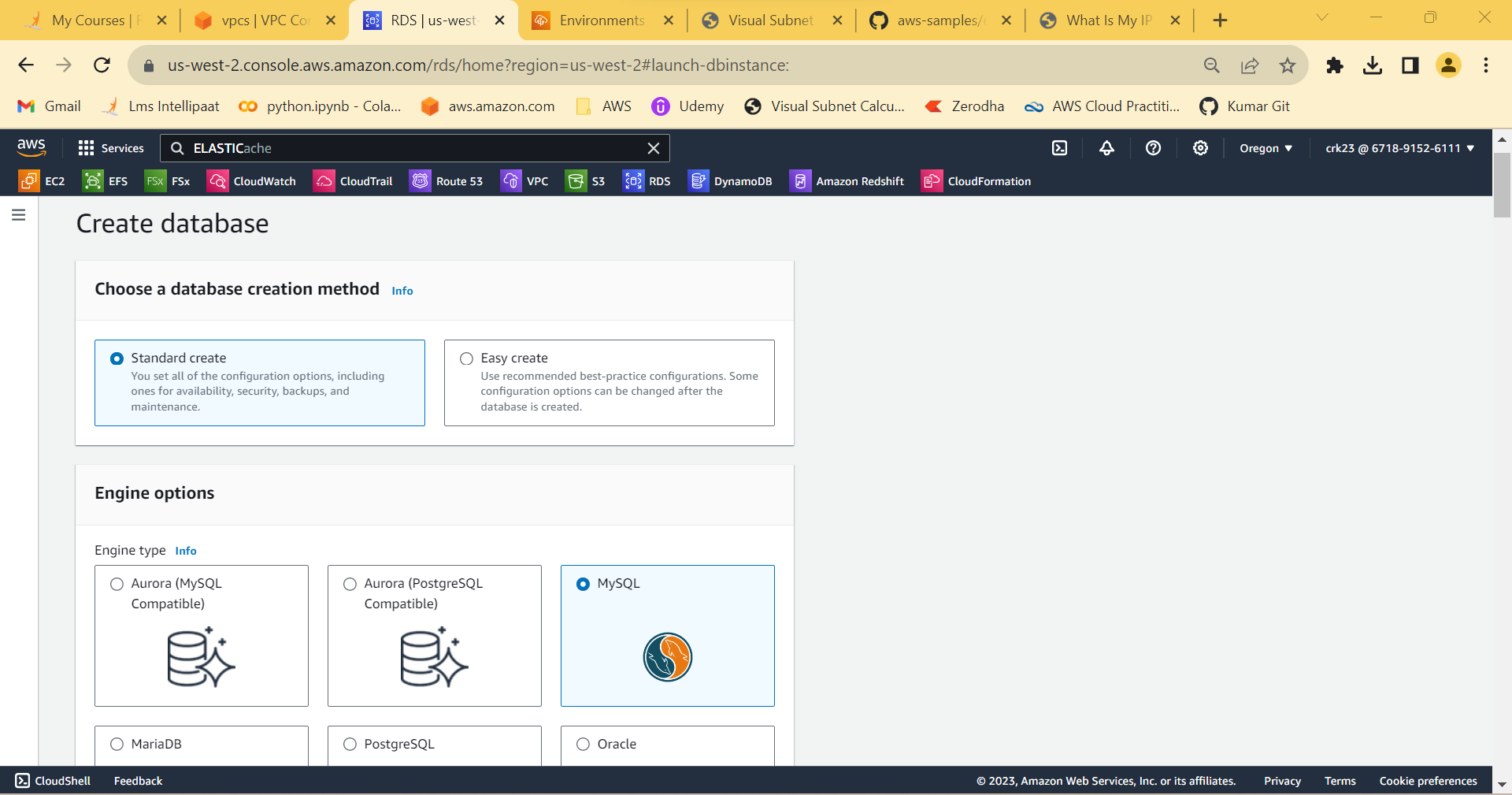
STEP 14 : Choose Upload and Deployand select a file

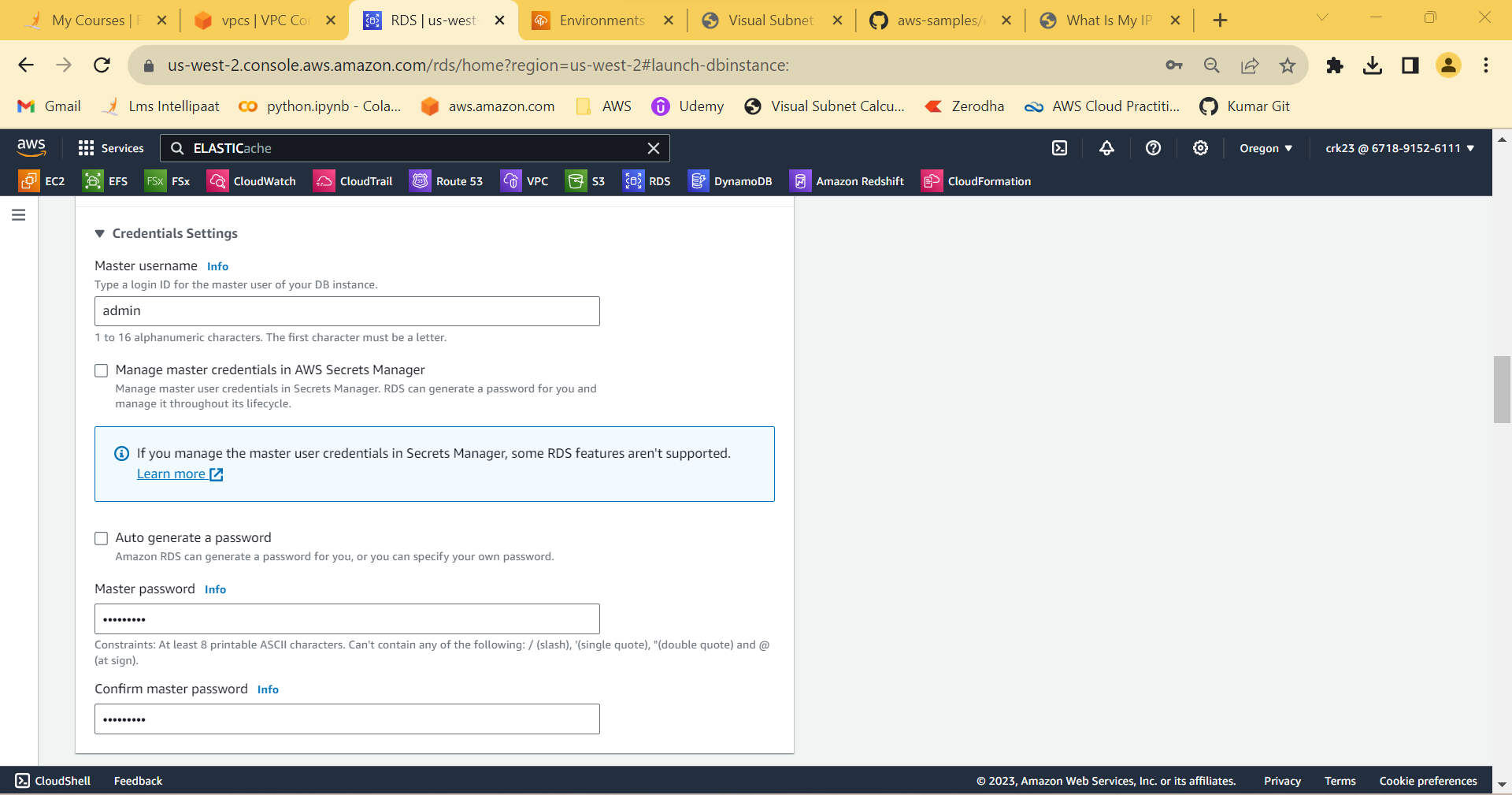
STEP 15 : Environment is working

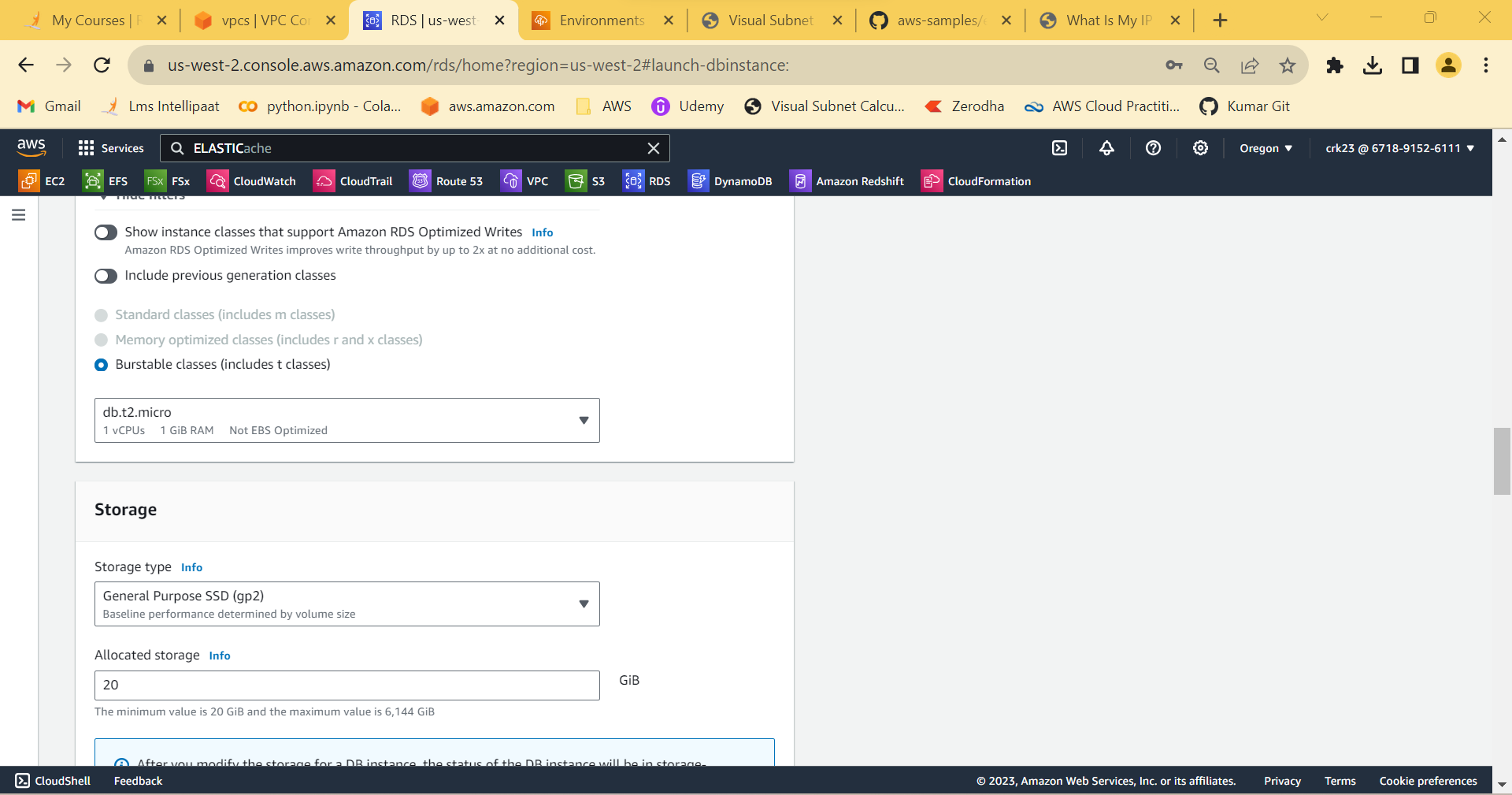
STEP 16 : Select actions > terminate environment

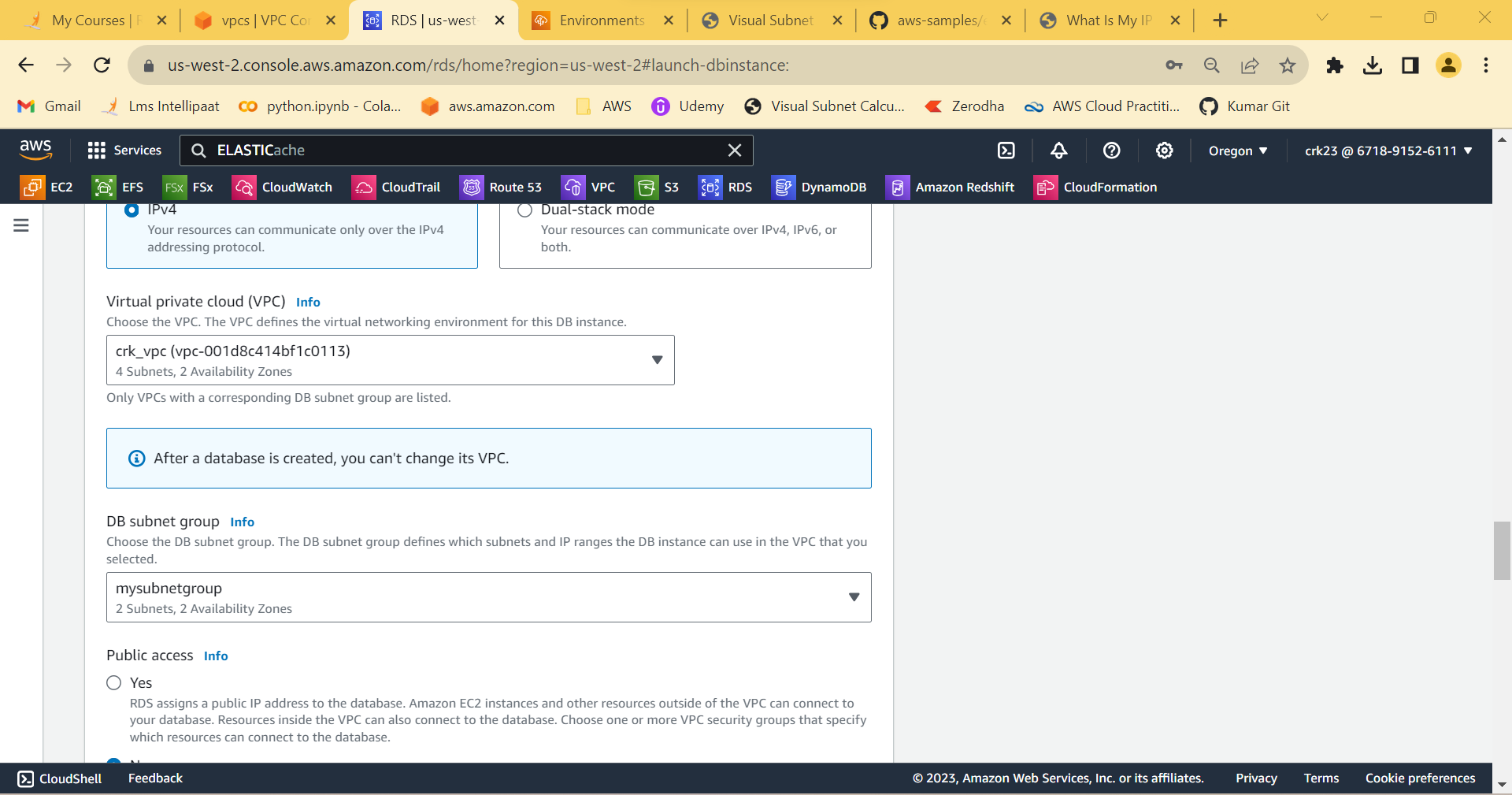
STEP 17 : Terminnate the RDS instance also.

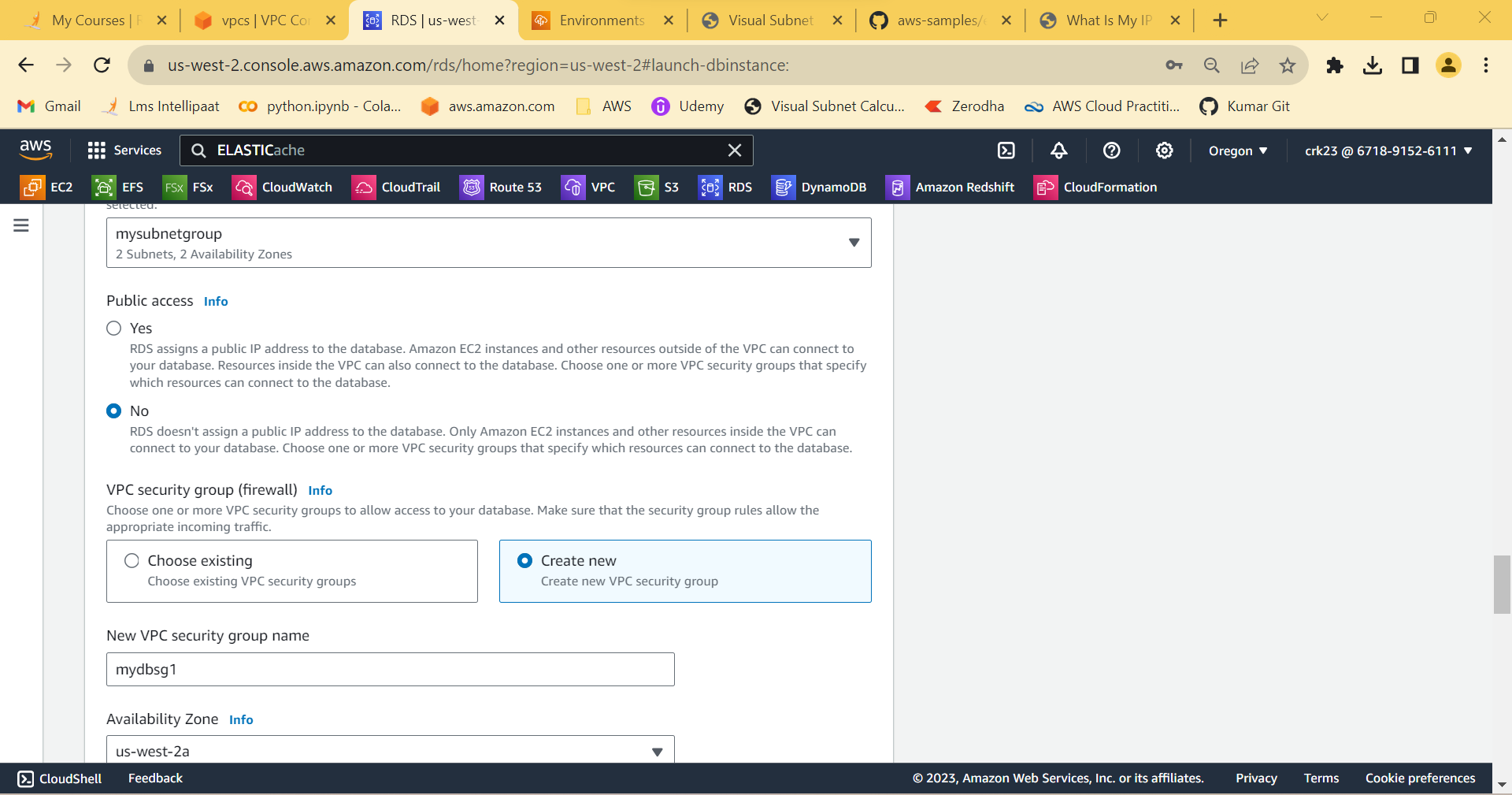


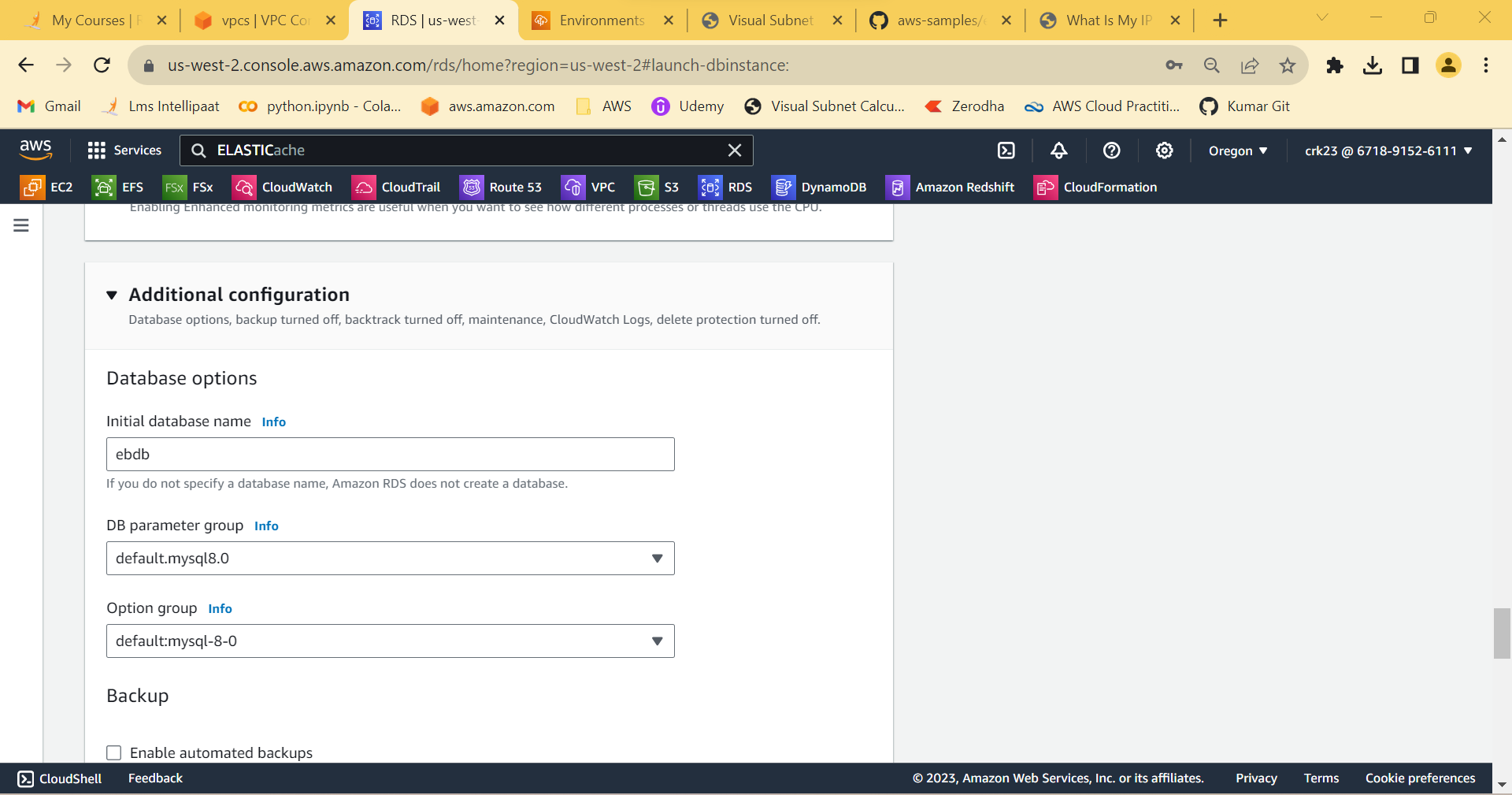


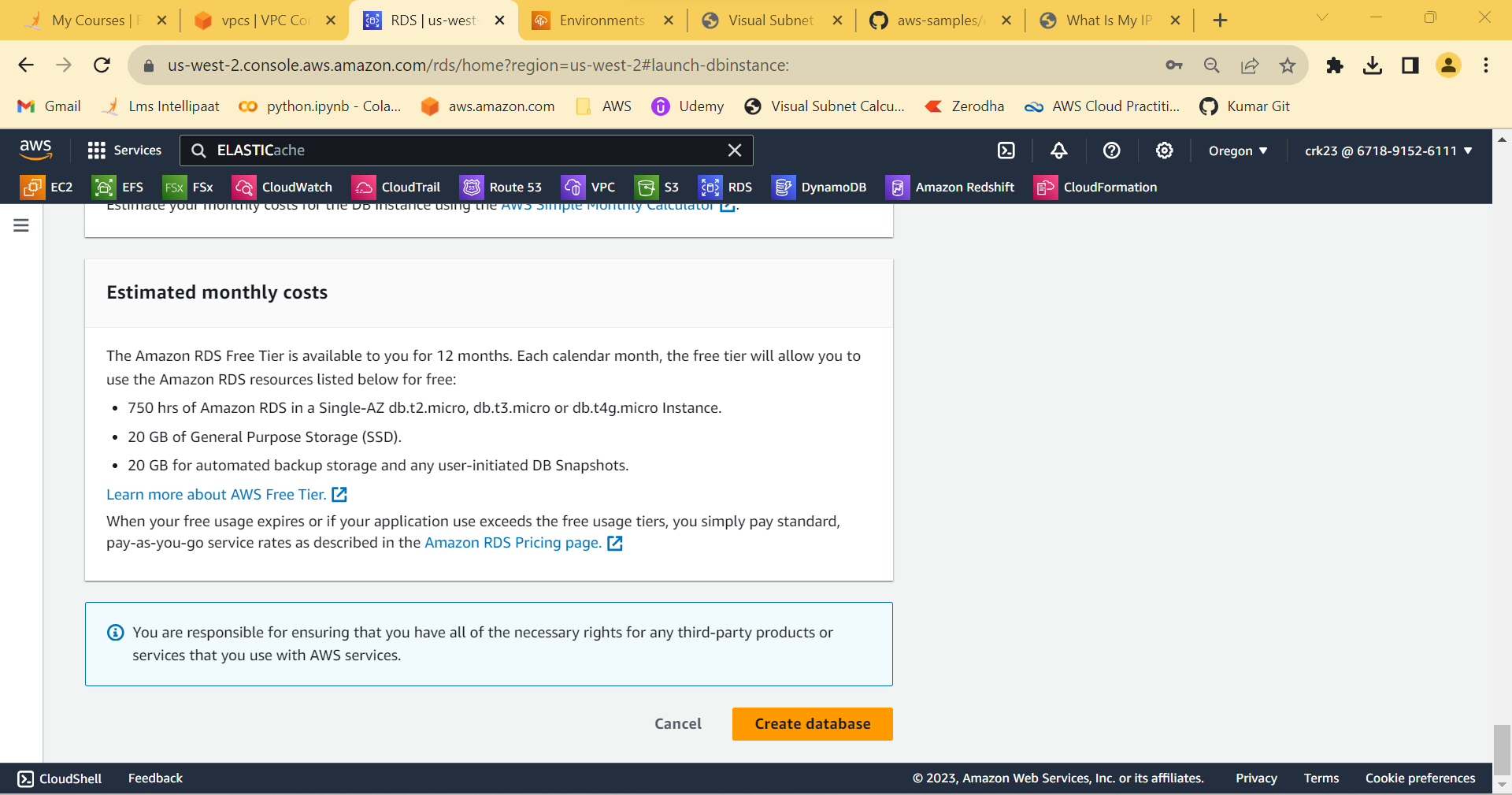


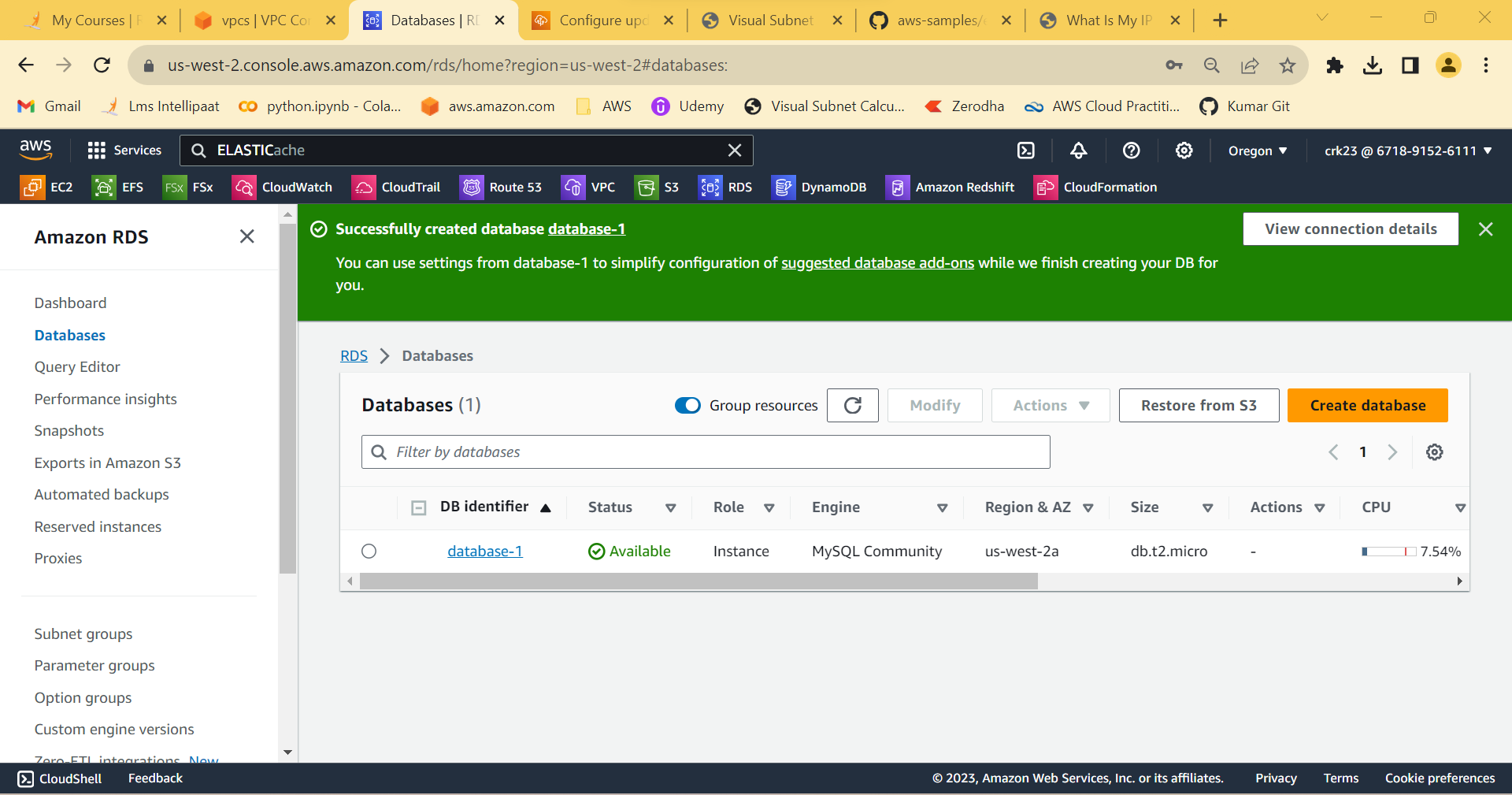


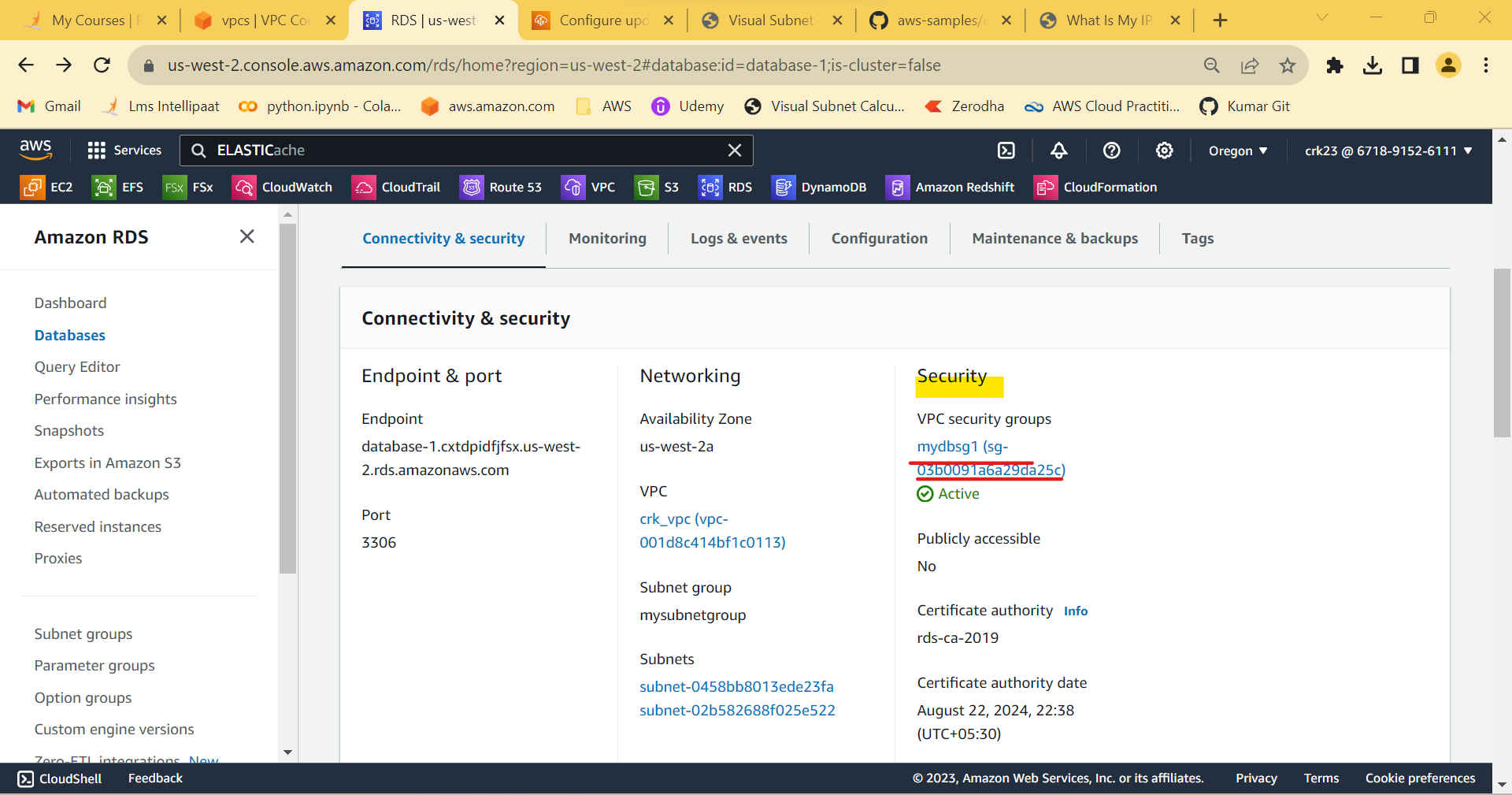


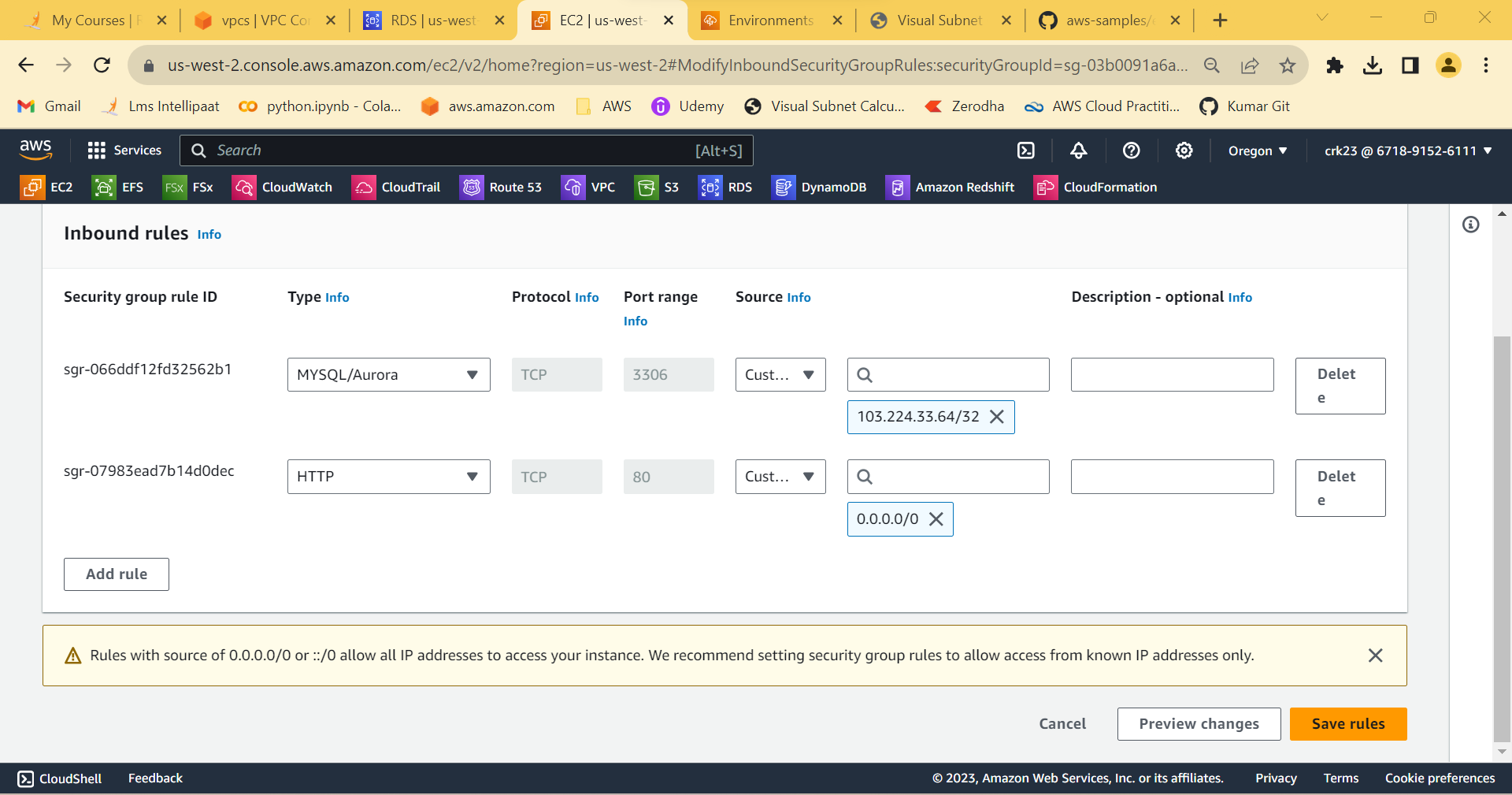


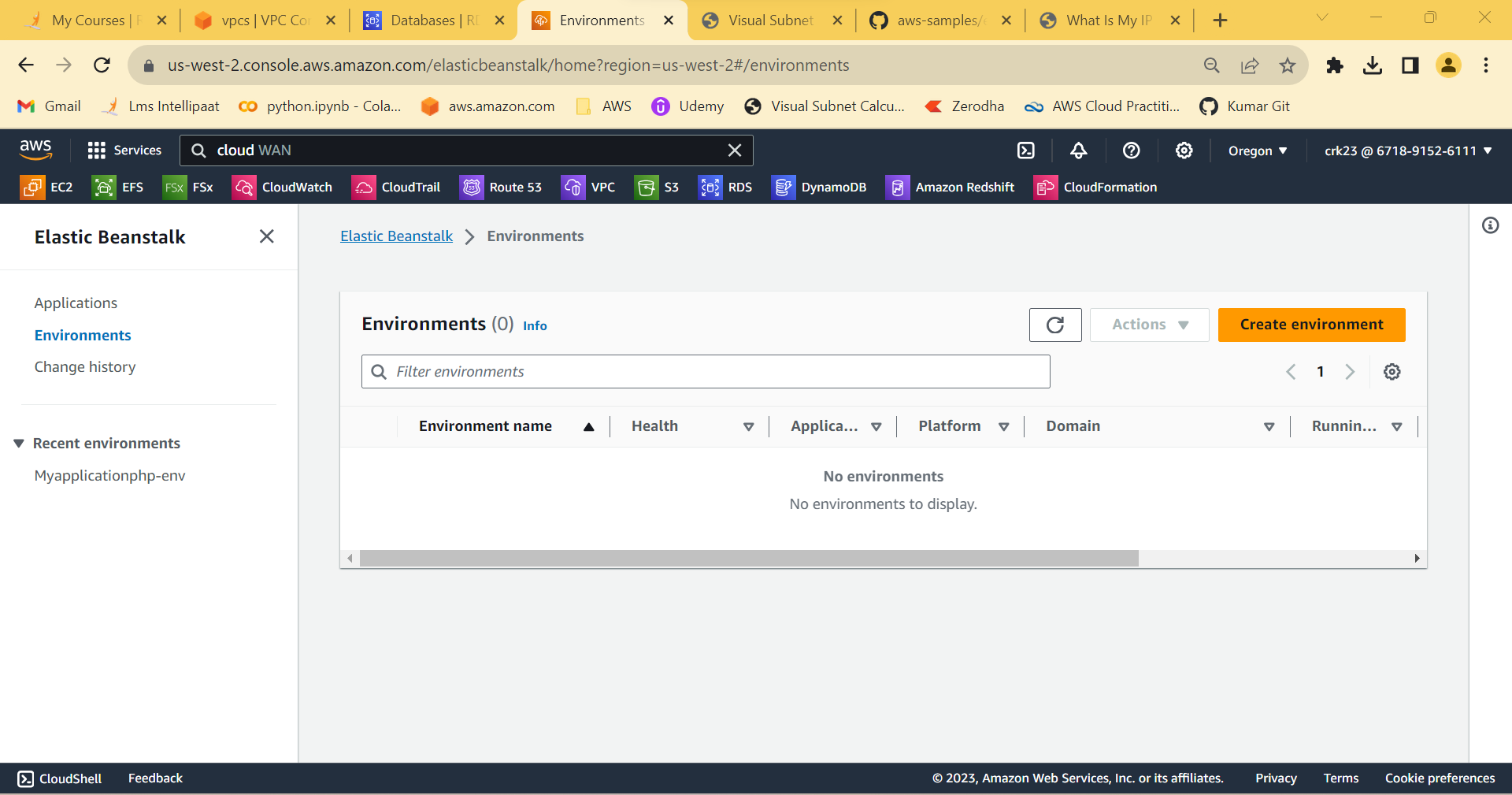


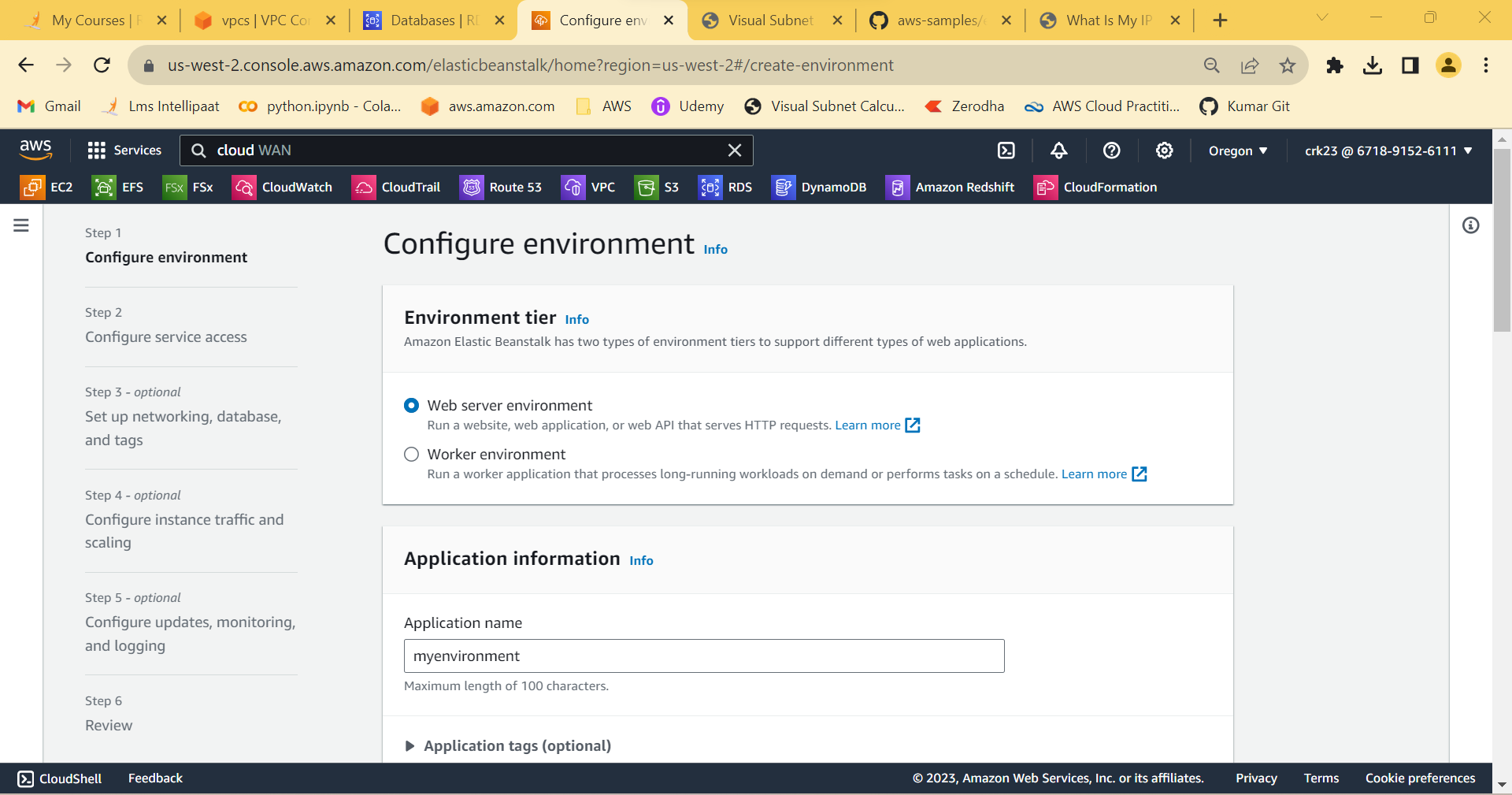


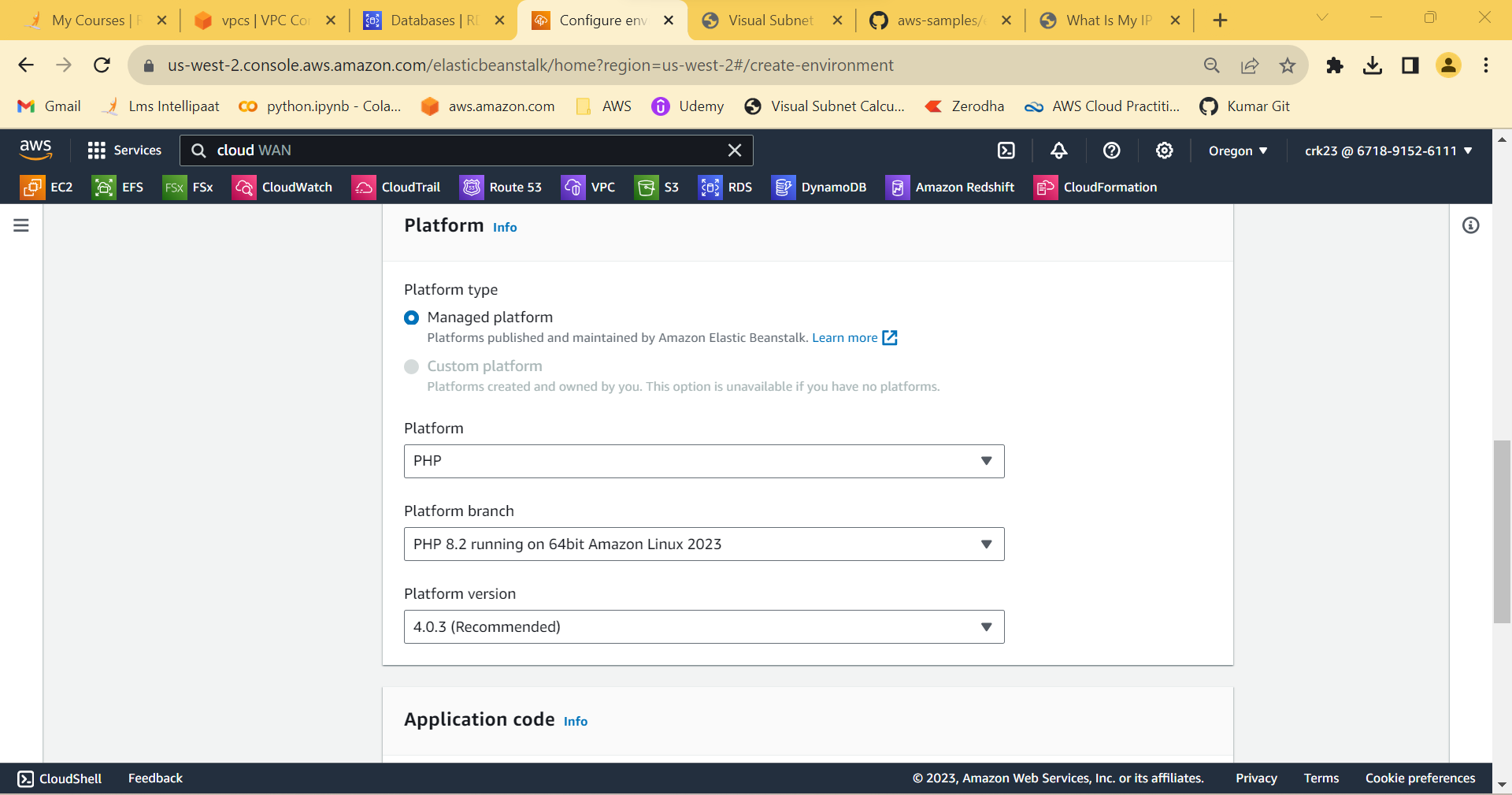


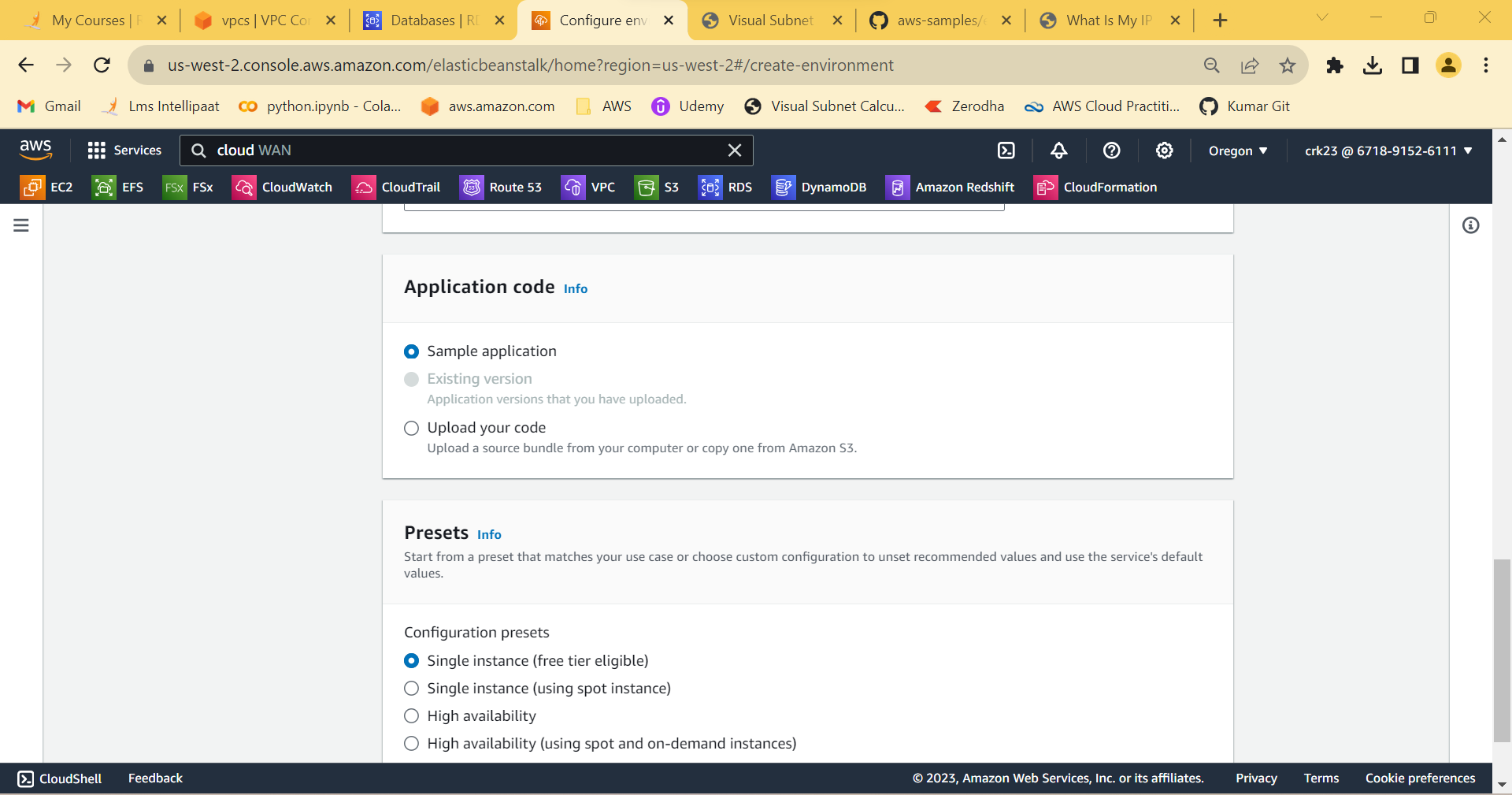


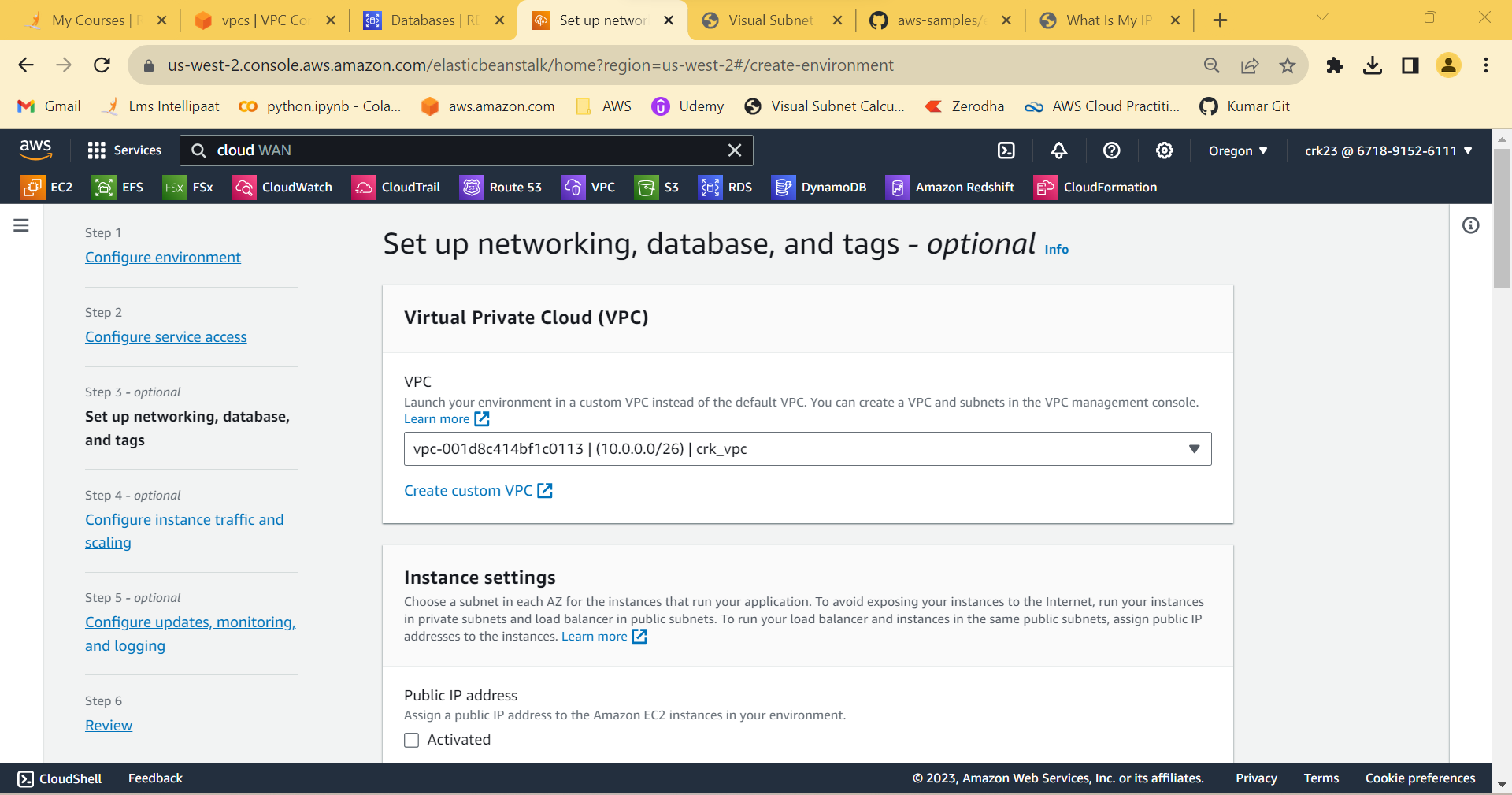


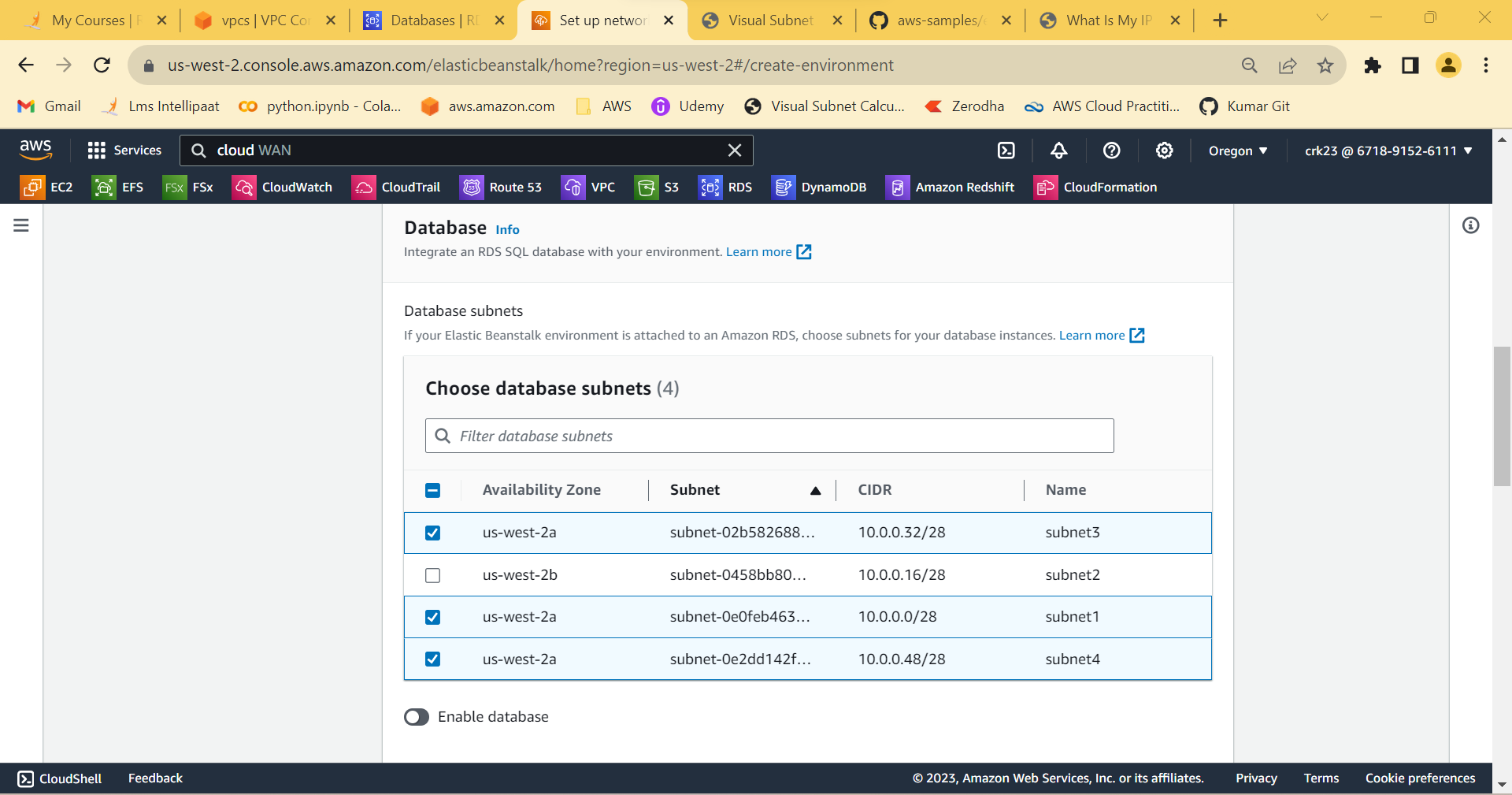


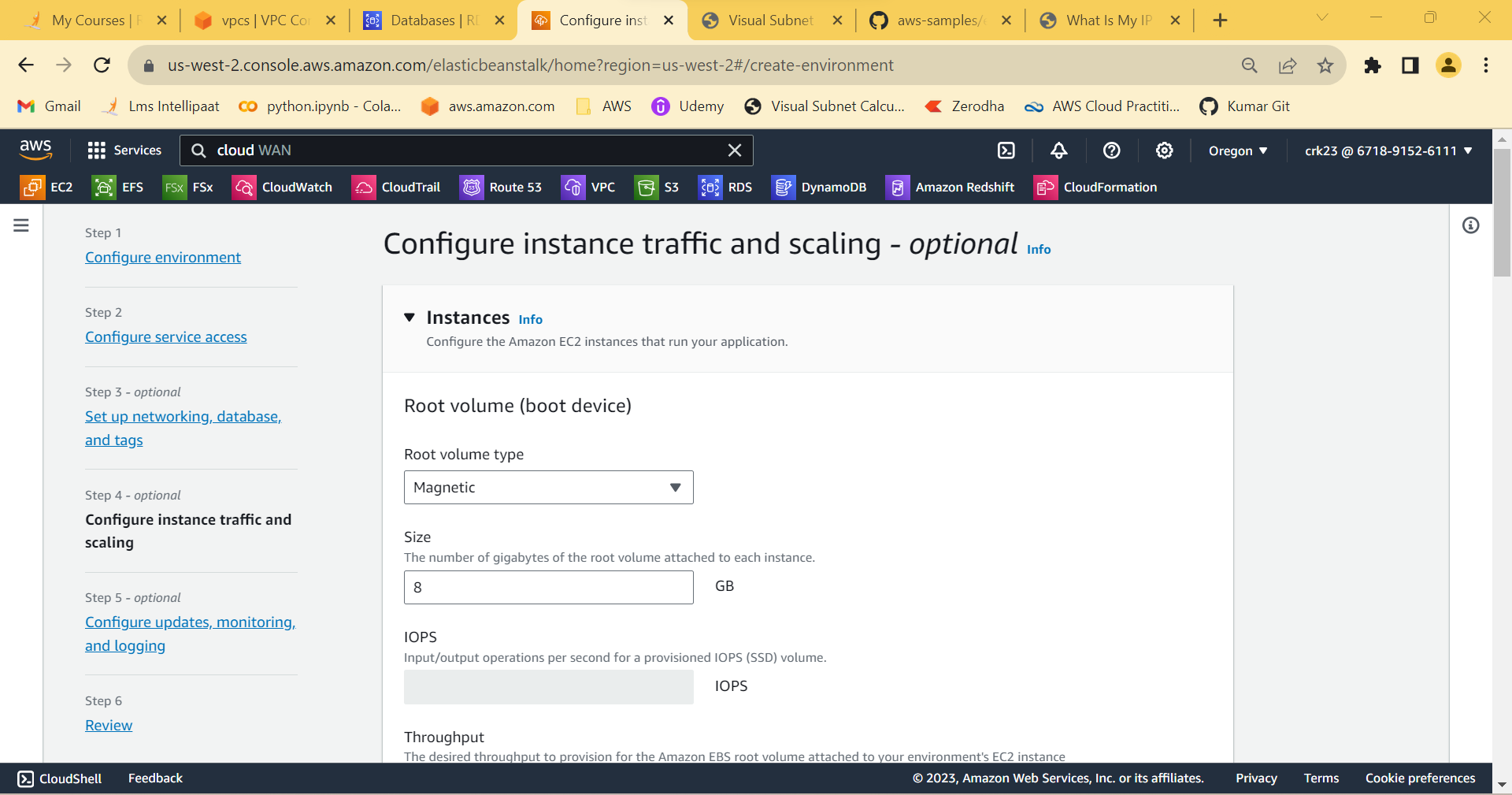


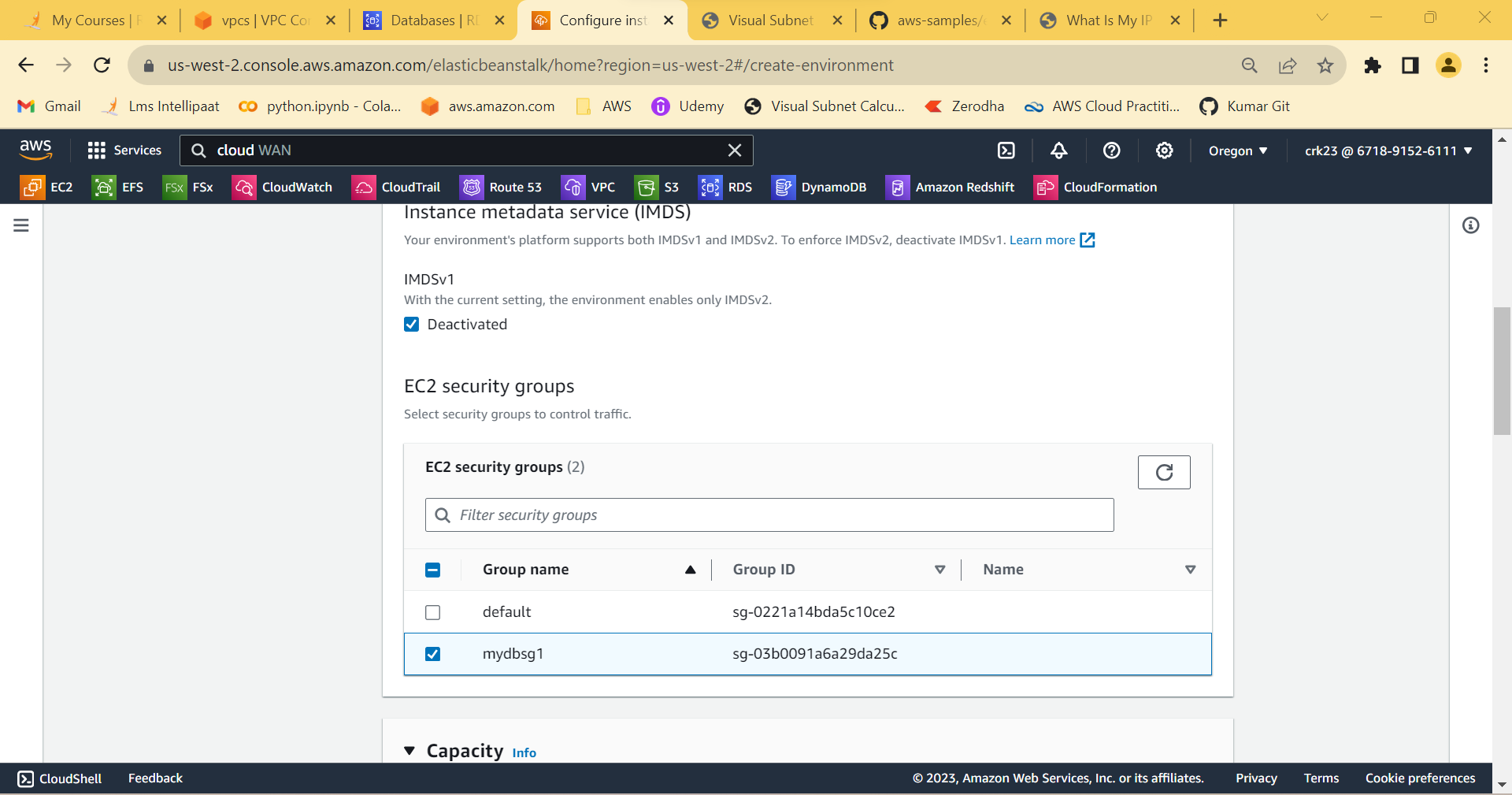


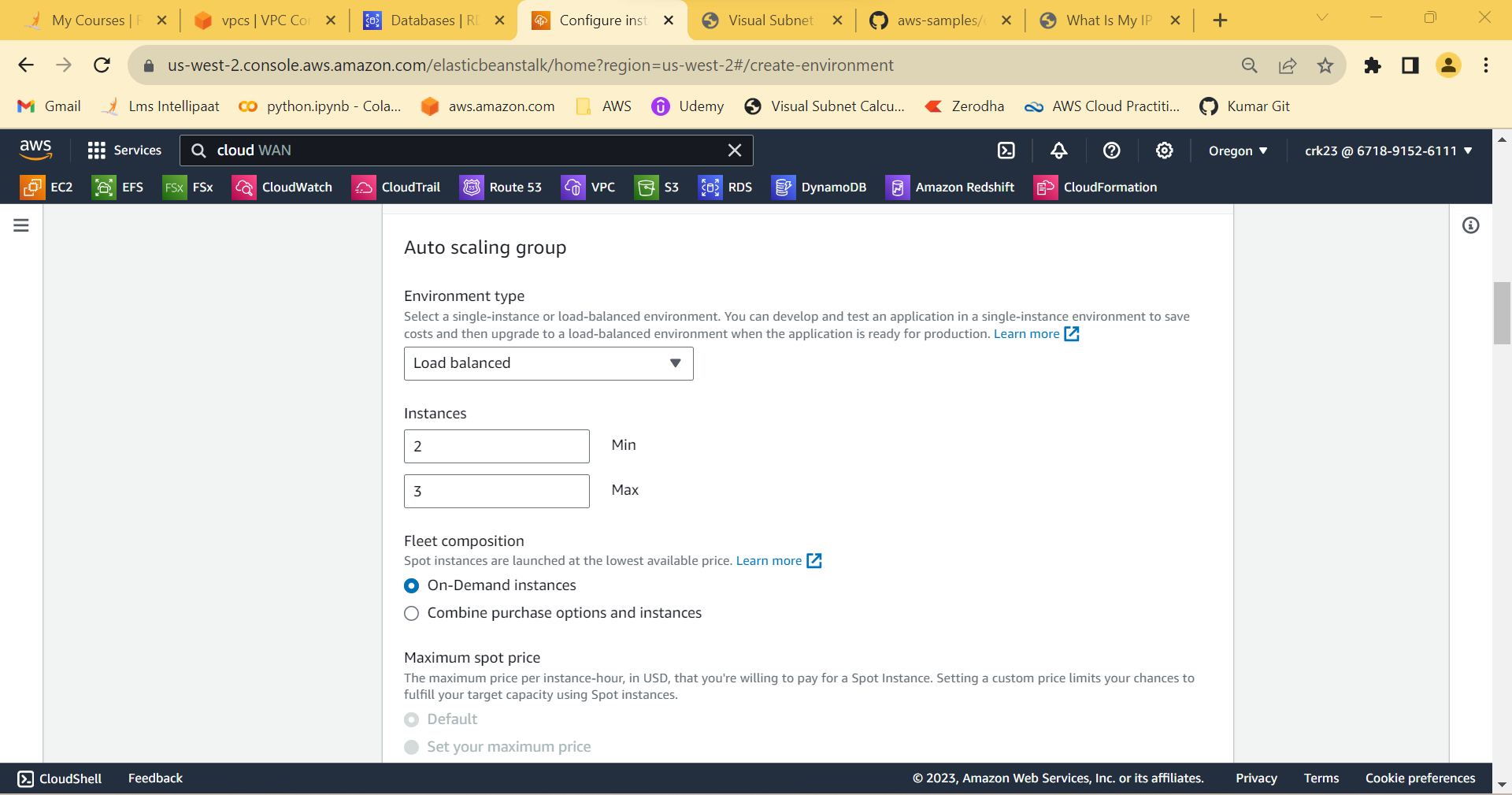


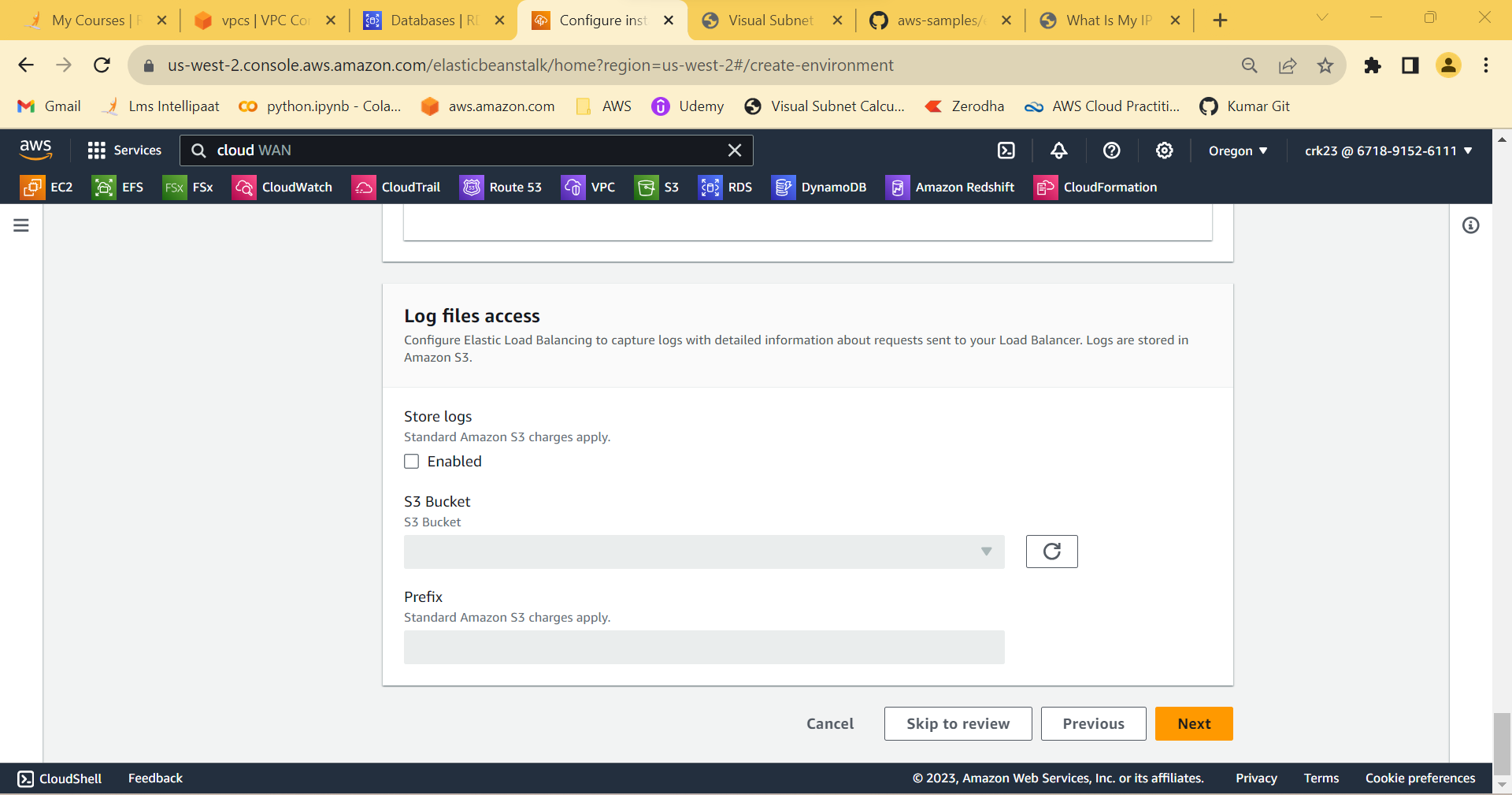


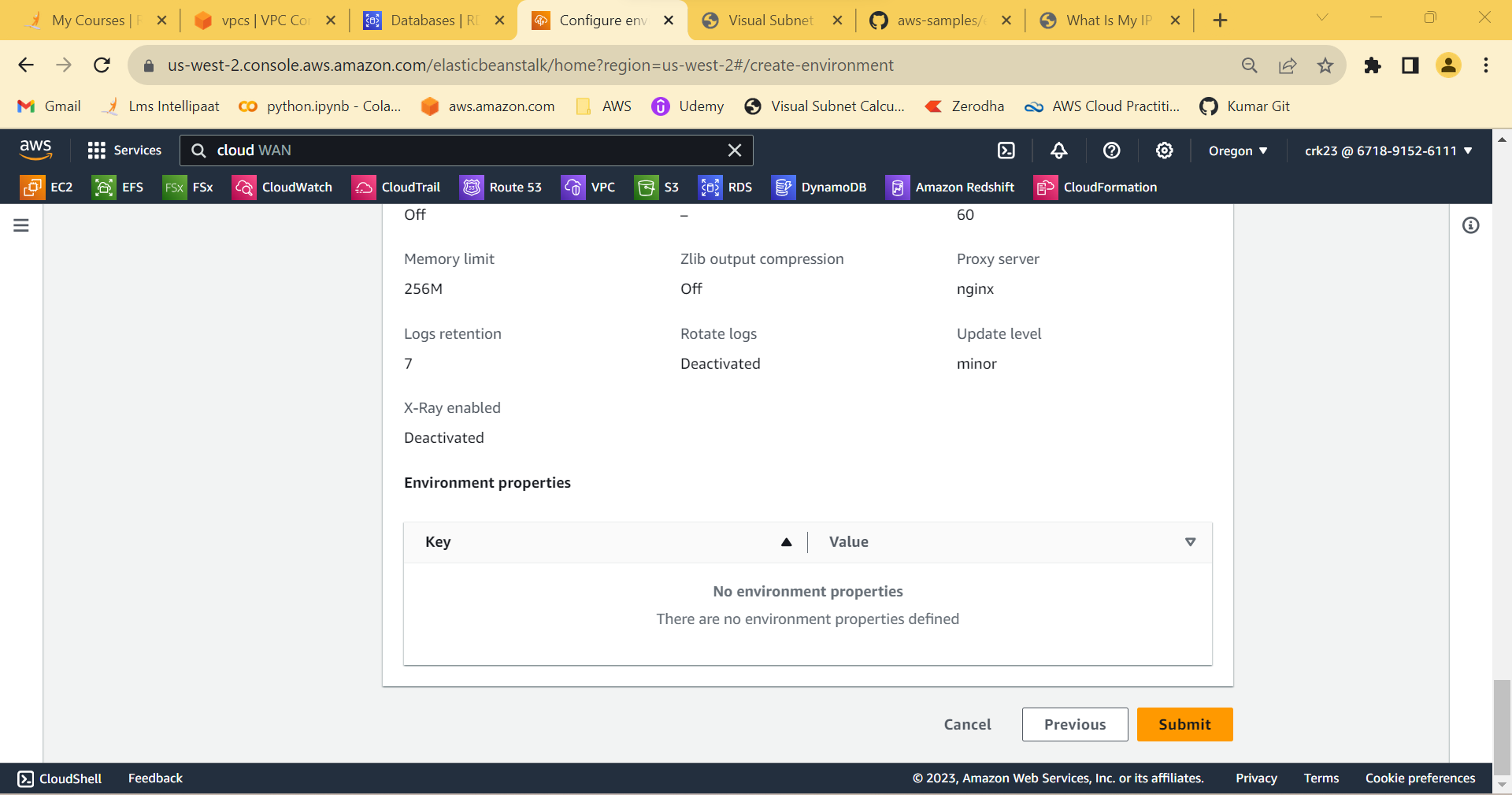


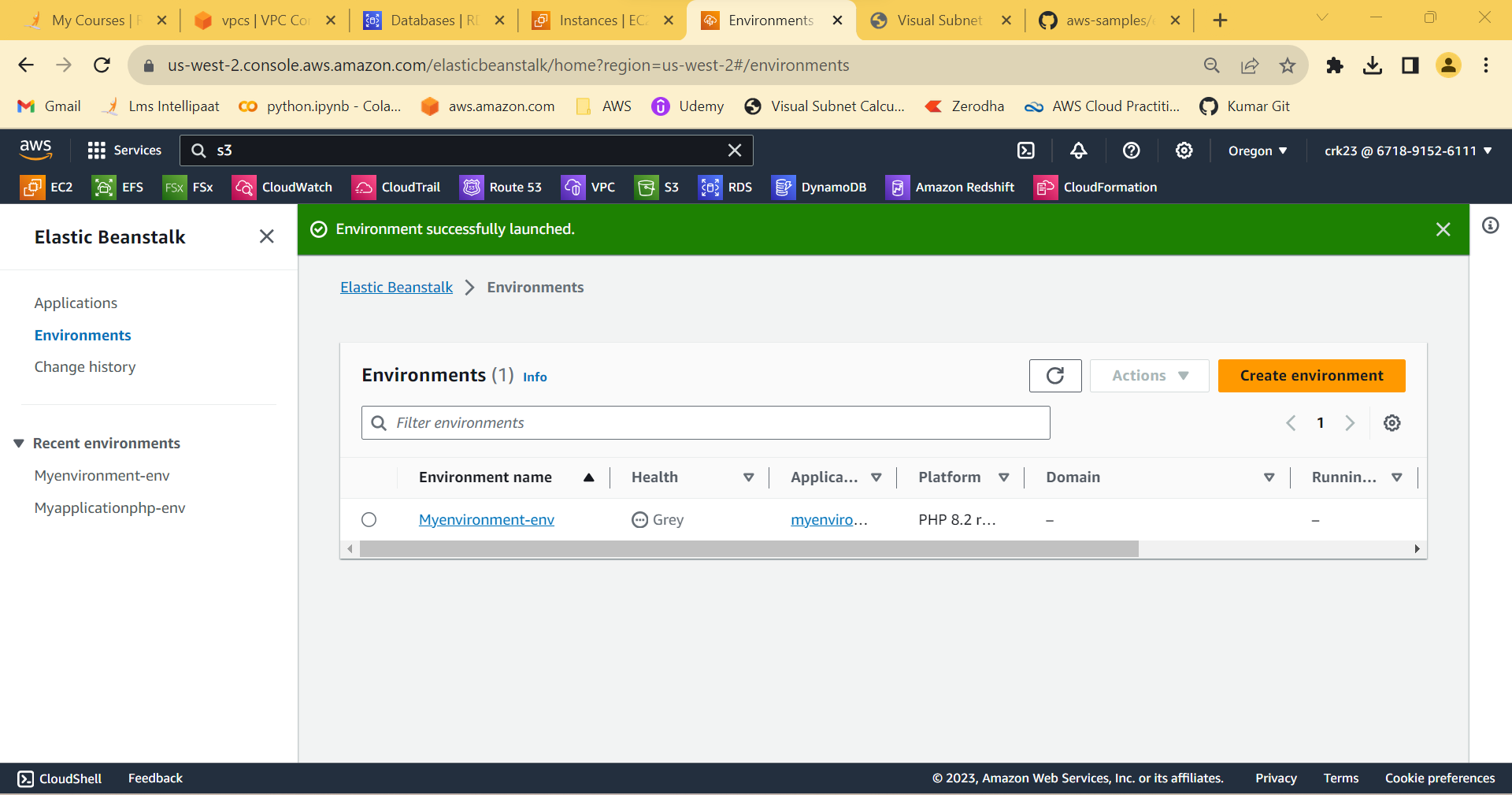


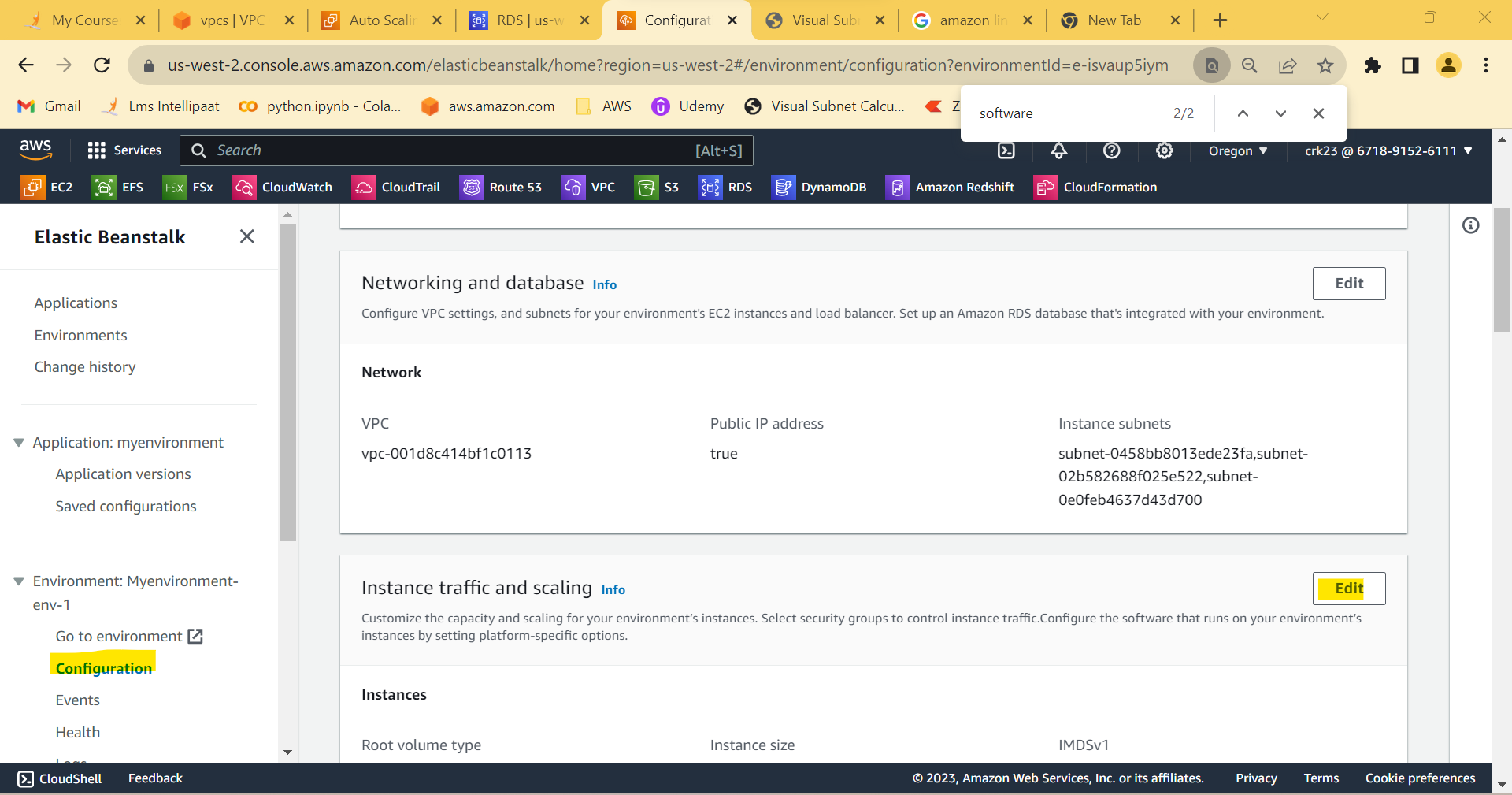


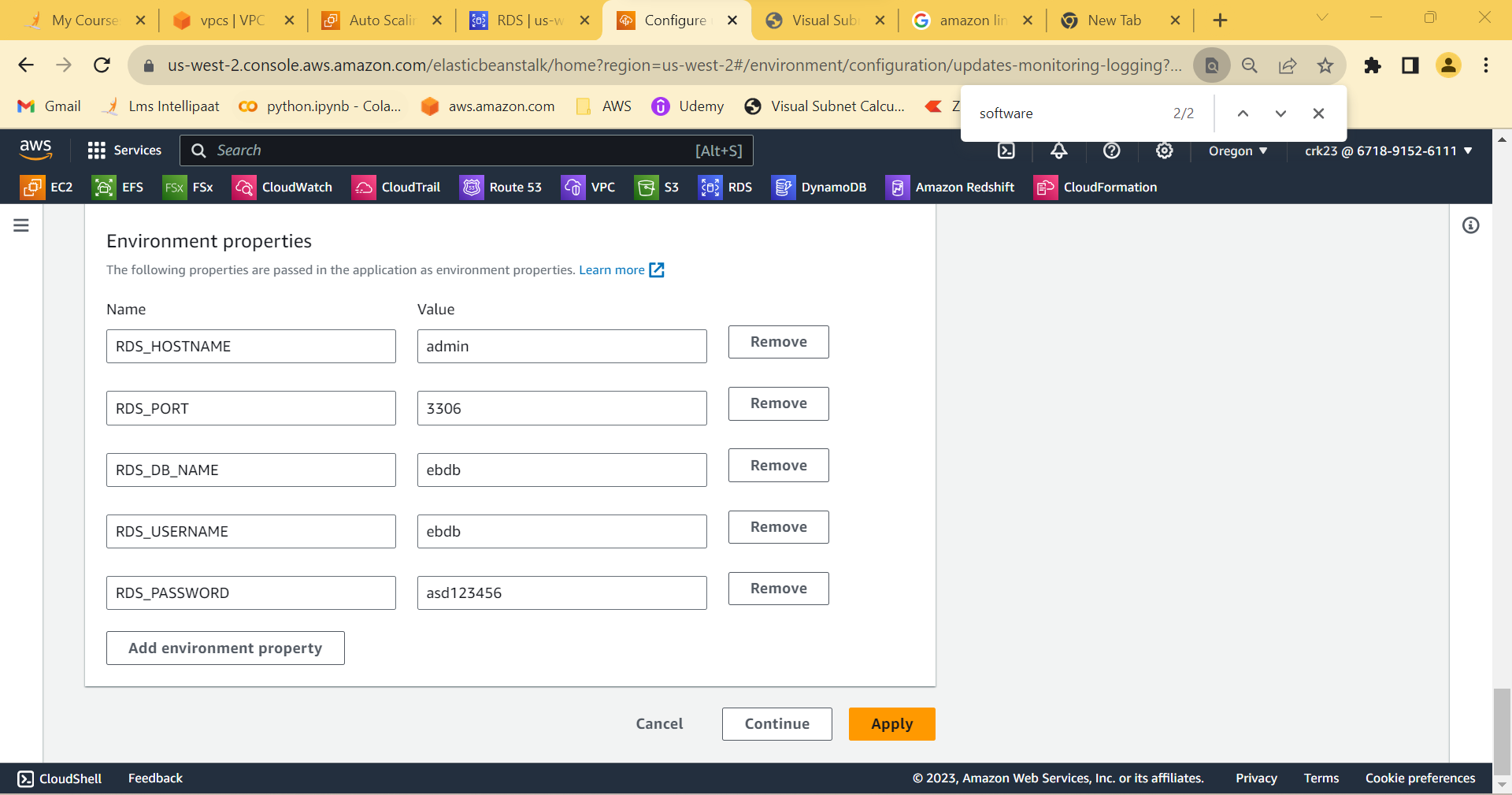


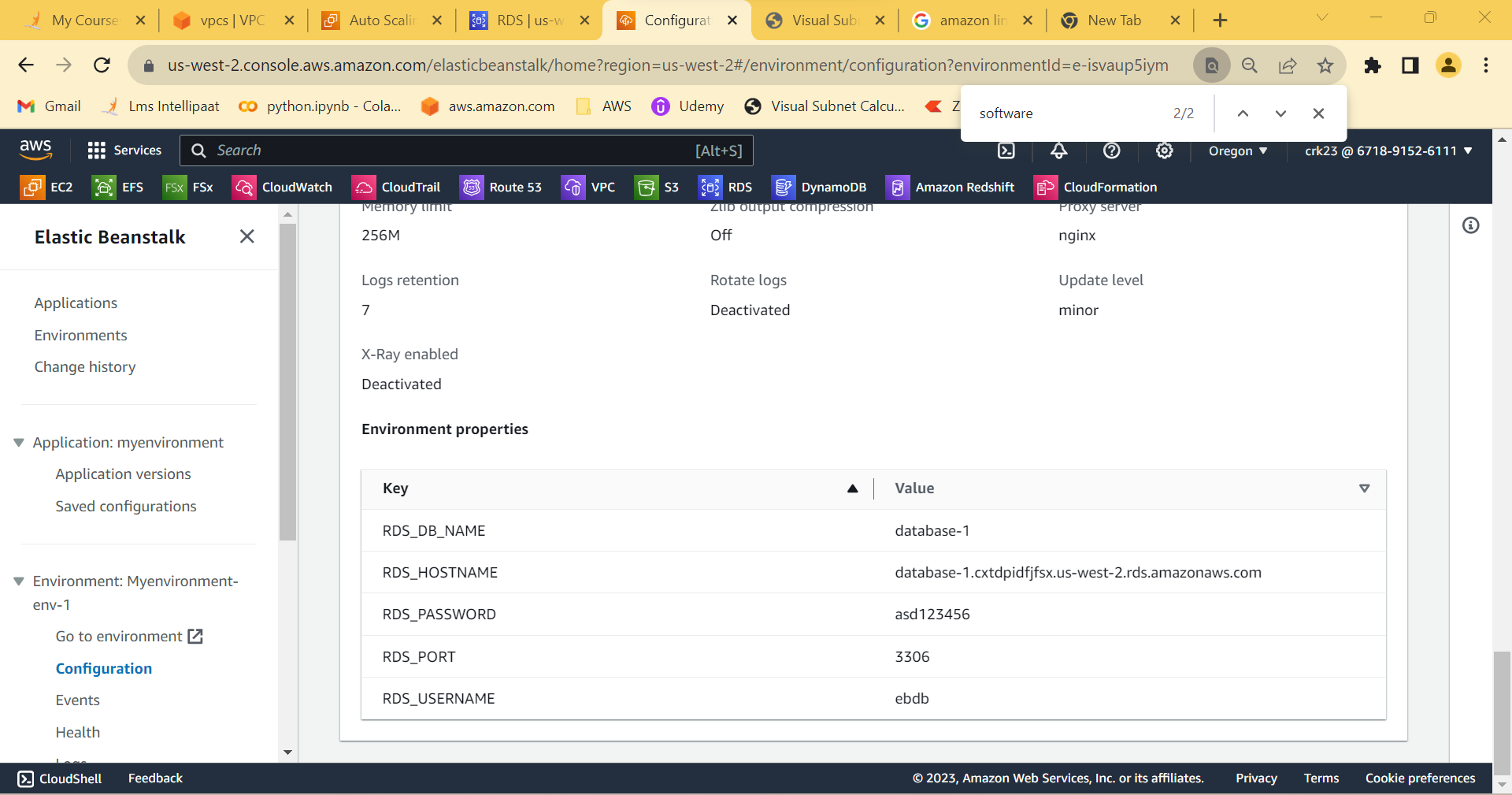


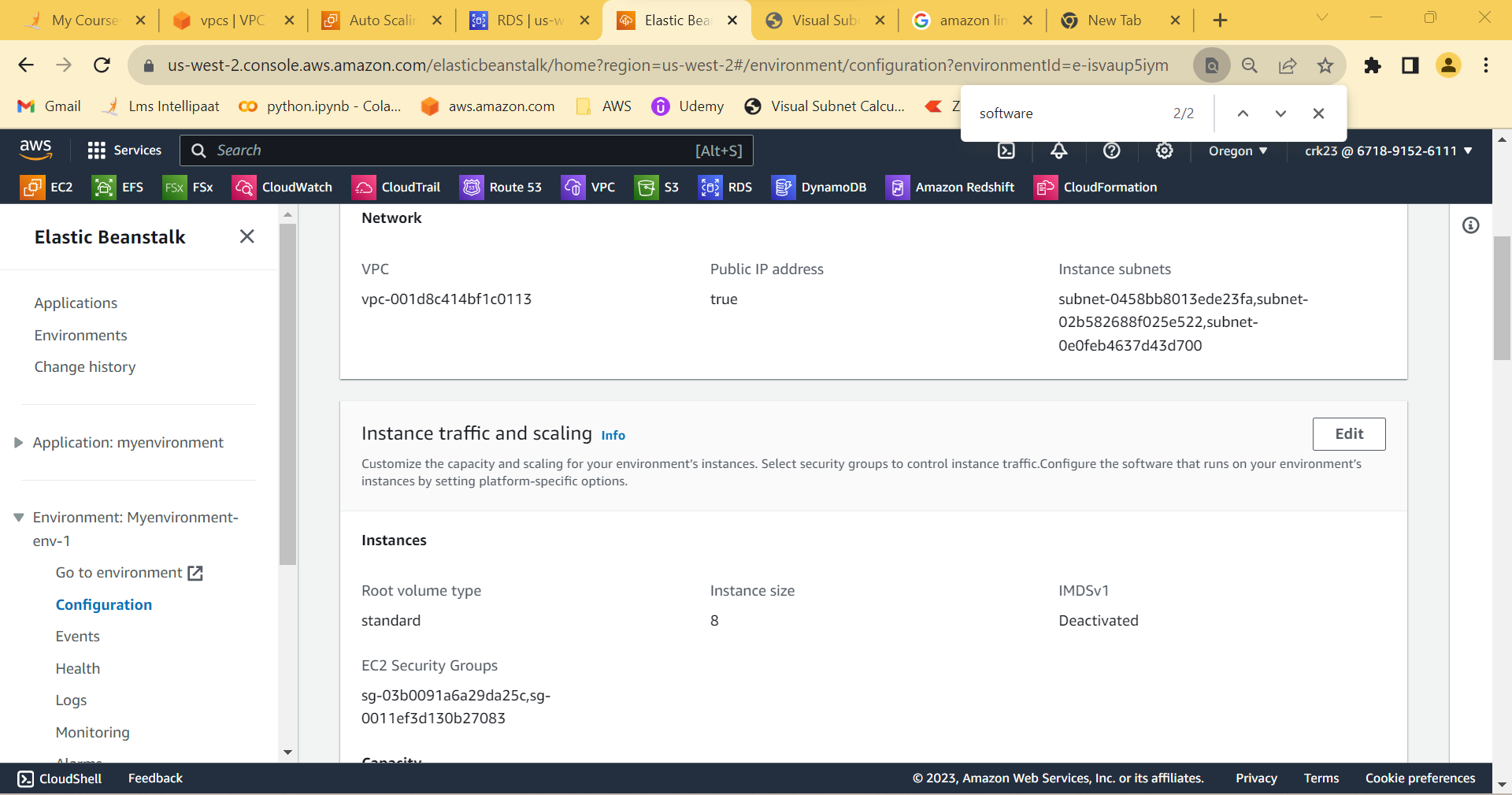


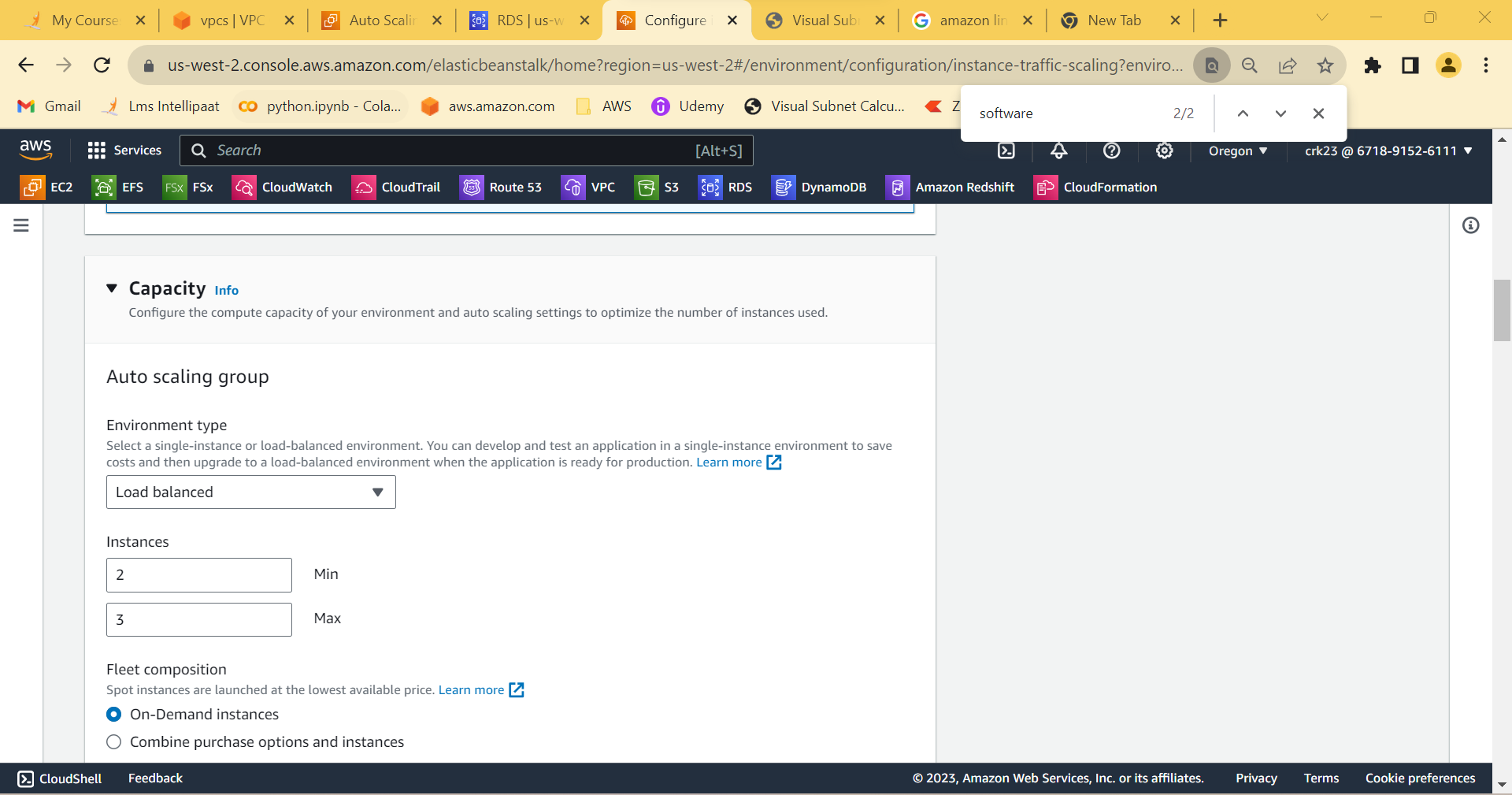


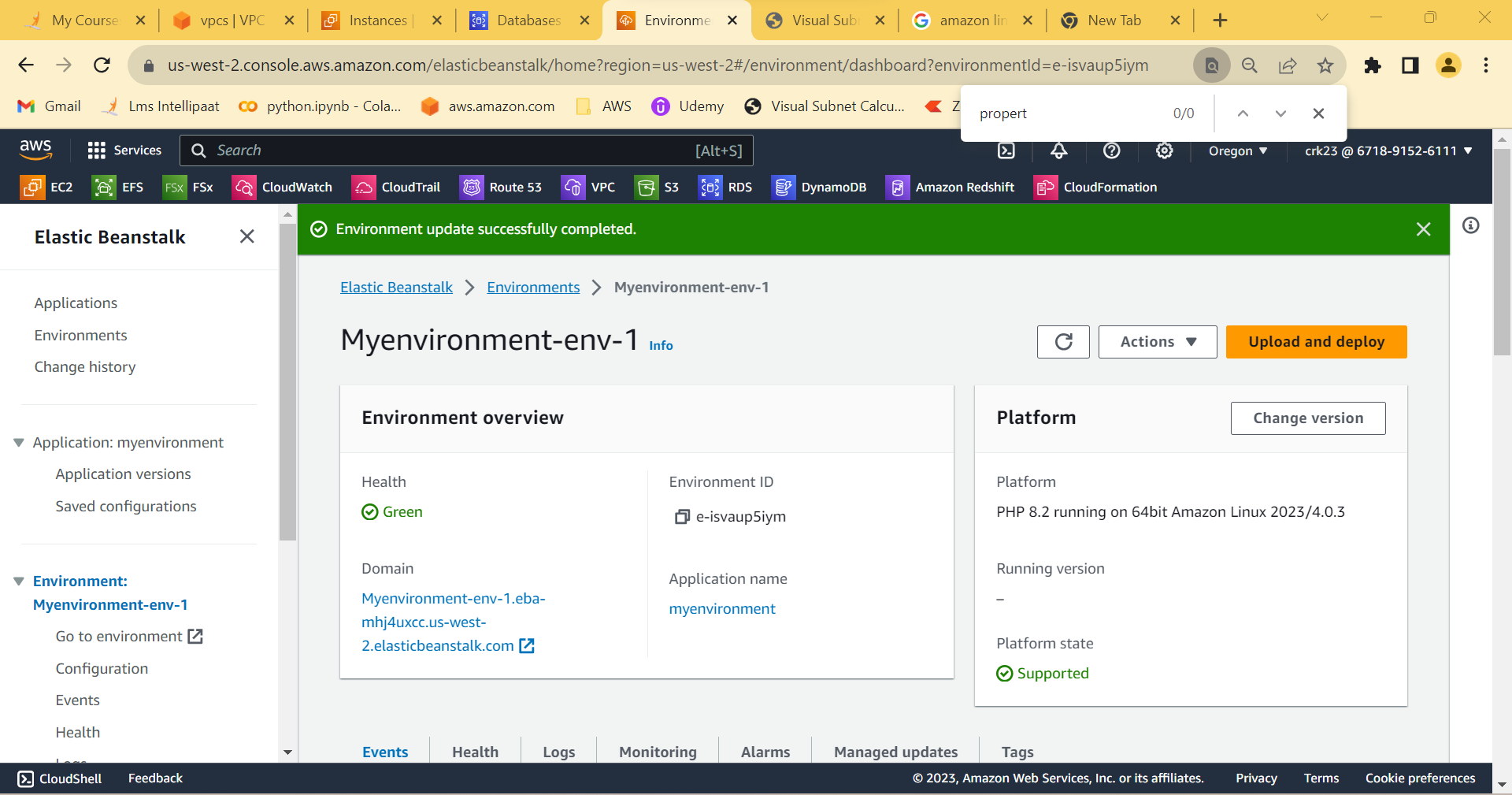


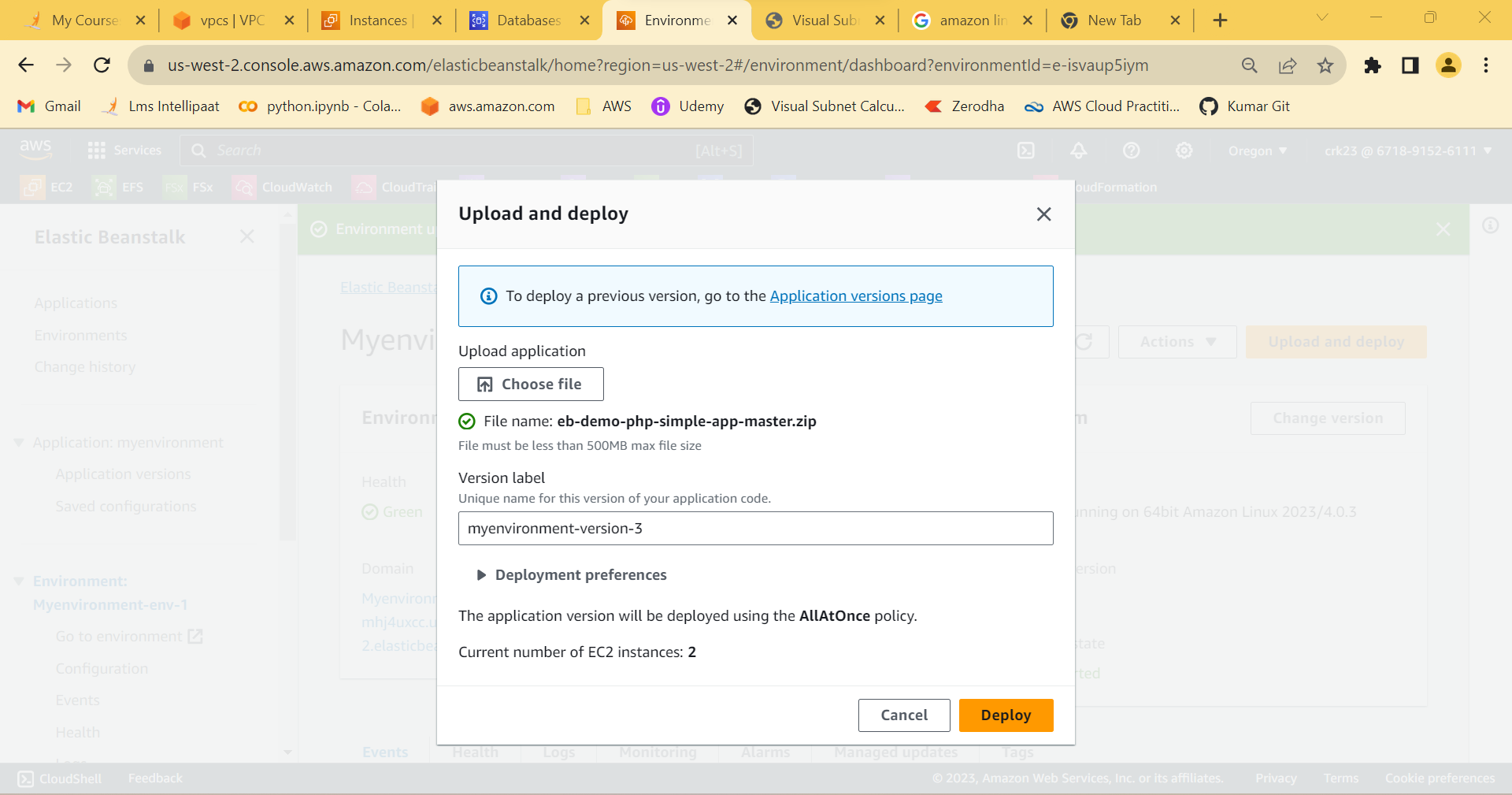


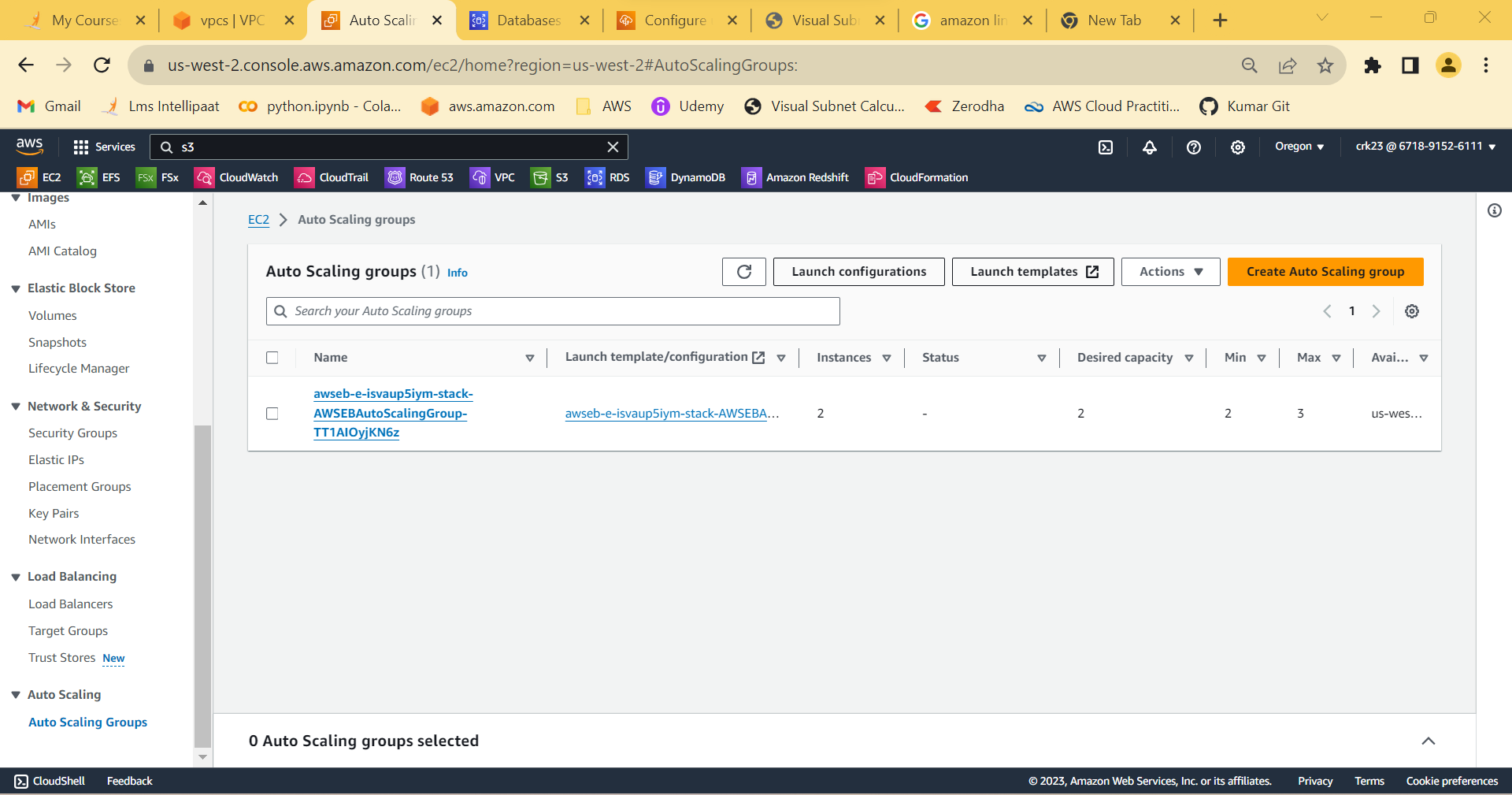












**OUTPUT:**

Created an DB instance, Created a ElasticBeanstalk environment and configured security groups and scaling.